

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re U.S. Patent Application of )  
OKUDA )  
Application Number: To Be Assigned )  
Filed: Concurrently Herewith )  
For: SEMICONDUCTOR INTEGRATED CIRCUIT DEVICE, )  
DATA PROCESSING SYSTEM AND MEMORY )  
SYSTEM )  
ATTORNEY DOCKET NO. HITA.0652 )

Honorable Assistant Commissioner  
for Patents  
Washington, D.C. 20231

**REQUEST FOR PRIORITY  
UNDER 35 U.S.C. § 119  
AND THE INTERNATIONAL CONVENTION**

Sir:

In the matter of the above-captioned application for a United States patent, notice is hereby given that the Applicant claims the priority date of the Japanese Patent Application 2002-211973, the filing date which is July 22, 2002, from the PCT application PCT/JP03/08973.

Respectfully submitted,

Stanley P. Fisher  
Registration Number 24,344

  
Juan Carlos A. Marquez  
Registration No. 34,072

**REED SMITH LLP**  
3110 Fairview Park Drive  
Suite 1400  
Falls Church, Virginia 22042  
(703) 641-4200  
December 20, 2003

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

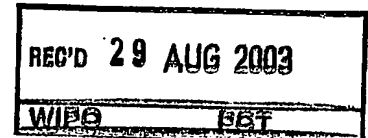
15.07.03

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日  
Date of Application: 2002年 7月22日

出 願 番 号  
Application Number: 特願2002-211973  
[ST. 10/C]: [JP 2002-211973]



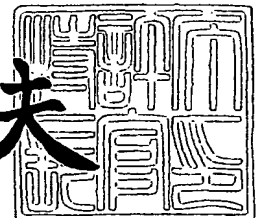
出 願 人  
Applicant(s): 株式会社日立製作所

PRIORITY DOCUMENT  
SUBMITTED OR TRANSMITTED IN  
COMPLIANCE WITH  
RULE 17.1(a) OR (b)

2003年 8月14日

特許庁長官  
Commissioner,  
Japan Patent Office

今 井 康 夫



Best Available Copy

【書類名】 特許願

【整理番号】 H02006411

【提出日】 平成14年 7月22日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 13/16

【発明者】

    【住所又は居所】 東京都小平市上水本町五丁目 2 0 番 1 号 株式会社日立  
                            製作所 半導体グループ内

    【氏名】 奥田 裕一

【特許出願人】

    【識別番号】 000005108

    【氏名又は名称】 株式会社日立製作所

【代理人】

    【識別番号】 100081938

    【弁理士】

    【氏名又は名称】 徳若 光政

    【電話番号】 0422-46-5761

【手数料の表示】

    【予納台帳番号】 000376

    【納付金額】 21,000円

【提出物件の目録】

    【物件名】 明細書 1

    【物件名】 図面 1

    【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 半導体集積回路装置、データ処理システム及びメモリシステム

【特許請求の範囲】

【請求項 1】 命令、情報、情報の所在位置、タイミング信号のいずれかを  
含む入力信号が供給される入力端子と、

上記入力信号に応答し、内部回路で形成された信号又は上記入力端子から供給  
された信号を出力させる出力端子とを備えてなることを特徴とする半導体集積回  
路装置。

【請求項 2】 請求項 1 において、

上記命令は、動作状態を指示するコマンドであり、

上記情報は、記憶すべきデータであり、

上記情報の所在位置は、アドレス信号であり、

上記タイミング信号は、クロックであり、

上記半導体集積回路装置は、上記クロックに同期して入力されたコマンド及び  
アドレス信号に対応して動作するメモリ回路を含むものであることを特徴とする  
半導体集積回路装置。

【請求項 3】 請求項 2 において、

上記メモリ回路は、自身に割り当てられたアドレスに対応した読み出し動作が  
指示されたときに、上記アドレスに従って読み出された記憶情報を入力端子の入  
力情報に置き換えて出力端子から出力するものであることを特徴とする半導体集  
積回路装置。

【請求項 4】 請求項 1 において、

上記出力端子から出力される命令、情報、情報の所在位置、タイミング信号は  
、内部で再生されたタイミング信号により再調整されるものであることを特徴と  
する半導体集積回路装置。

【請求項 5】 請求項 4 において、

上記再調整されるタイミング信号は、基準タイミング信号を受ける位相同期ル  
ープ回路で生成されるものであることを特徴とする半導体集積回路装置。

【請求項 6】 請求項 5 において、

上記基準タイミング信号は、外部から入力されるものであることを特徴とする半導体集積回路装置。

【請求項 7】 命令、情報、情報の所在位置、タイミング信号のいずれかを含み入力信号が供給される入力端子と、かかる入力信号に応答し、内部回路で形成された信号又は上記入力端子から供給された信号を出力させる出力端子とを備えた複数の半導体集積回路装置と、

上記半導体集積回路装置に対して、命令、情報、情報の所在位置、タイミング信号のいずれかを含み入力信号を生成する信号生成回路とを備え、

前段とされる半導体集積回路装置の出力端子と次段とされる半導体集積回路装置の入力端子との対応するもの同士が接続され縦列形態にされ、

上記信号生成回路で生成された命令、情報、情報の所在位置、タイミング信号のいずれかを含み入力信号は、上記縦列形態の初段とされる半導体集積回路装置の入力端子に供給され、

上記縦列形態の終段の半導体集積回路装置の出力端子の出力信号のうち、少なくとも情報に対応した信号が上記信号処理回路に伝えられるものであることを特徴とするデータ処理システム。

【請求項 8】 請求項 7 において、

上記命令は、動作状態を指示するコマンドであり、

上記情報は、記憶すべきデータであり、

上記情報の所在位置は、アドレス信号であり、

上記タイミング信号は、クロックであり、

上記複数の半導体集積回路装置の各々は、上記クロックに同期して入力されたコマンド及びアドレス信号に対応して動作するメモリ回路を含むものであることを特徴とするデータ処理システム。

【請求項 9】 請求項 8 において、

上記出力端子から出力されるコマンド、データ、アドレス及びタイミング信号は、内部で再生されたタイミング信号により再調整されるものであることを特徴とするメモリシステム。

【請求項 10】 コマンド、データ、アドレス、タイミング信号を含む入力

信号がそれぞれ供給される入力端子と、

上記入力端子から供給された入力信号に対応した信号をそれぞれ出力させる出力端子とを含む半導体記憶装置の複数個を備え、

上記複数の半導体記憶装置のうち、前段とされる半導体記憶装置の出力端子と次段とされる半導体記憶装置の入力端子との対応するもの同士が接続されて縦列形態とされてなることを特徴とするメモリシステム。

【請求項 11】 請求項 10 において、

上記縦列形態の初段の半導体記憶装置の入力端子には、信号生成回路により形成されたコマンド、データ、アドレス、タイミング信号が供給され、

上記縦列形態の終段の半導体記憶装置の出力端子から出力される出力信号のうち、少なくともデータに対応した信号は、上記信号生成回路に伝えられるものであることを特徴とするメモリシステム。

【請求項 12】 請求項 11 において、

上記信号生成回路は、1つの半導体集積回路装置から構成されるメモリ制御装置であることを特徴とするメモリシステム。

【請求項 13】 請求項 12 において、

上記出力端子から出力されるコマンド、データ、アドレス及びタイミング信号は、内部で再生されたタイミング信号により再調整されるものであることを特徴とするメモリシステム。

【請求項 14】 請求項 13 において、

上記再調整されるタイミング信号は、基準タイミング信号を受ける位相同期ループ回路で生成されるものであることを特徴とするメモリシステム。

【請求項 15】 請求項 12 において、

上記メモリ制御装置により生成されたデータは、上記初段を構成する複数の半導体記憶装置の入力端子に振り分けられて供給され、

上記初段の複数の半導体記憶装置に対応して次段から終段まで複数の半導体記憶装置が設けられて、それぞれが一对一に対応して縦列接続されるものであることを特徴とするメモリシステム。

【請求項 16】 請求項 15 において、

上記信号生成回路で形成されたコマンド及びアドレスは、上記初段を構成する複数の半導体記憶装置の入力端子に共通に伝えられ、

上記初段の複数の半導体記憶装置の出力端子から終段の複数の半導体記憶装置の入力端子までの接続においては、上記コマンド及びアドレスが上記データに対応してそれぞれが一对一に接続されることを特徴とするメモリシステム。

【請求項 17】 請求項 15 において、

上記信号生成回路は、上記初段を構成する複数の半導体記憶装置の入力端子に対応した複数組のコマンド及びアドレスを生成し、上記初段を構成する複数の半導体記憶装置の入力端子に対して一对一に対応して伝えられ、

上記初段の複数の半導体記憶装置の出力端子から終段の複数の半導体記憶装置の入力端子までの接続においても、上記コマンド及びアドレスが上記データに対応してそれぞれが一对一に接続されることを特徴とするメモリシステム。

【請求項 18】 請求項 15 において、

上記複数の半導体記憶装置は、自身に割り当てられたアドレスに対応した読み出し動作が指示されたときに、上記アドレスに従って読み出された記憶情報を入力端子の入力情報に置き換えて出力端子から出力するものであることを特徴とするメモリシステム。

【請求項 19】 請求項 16 において、

上記縦列接続される前段の半導体記憶装置と後段の半導体記憶装置とは、共通の実装基板の表面と裏面に実装され、

スルーホールによって上記縦列形態に接続されるものであることを特徴とするメモリシステム。

【請求項 20】 請求項 10 において、

上記半導体記憶装置は、読み出し用バッファ回路を備え、

メモリセルから上記読み出し用バッファ回路に記憶情報を読み出して保持させる第 1 読み出しコマンドと、

上記読み出し用バッファ回路に保持された記憶情報を出力端子から出力させる第 2 読み出しコマンドを備えるものであることを特徴とするメモリシステム。

【請求項 21】 請求項 20 において、

上記半導体記憶装置は、書き込み用バッファ回路を備え、

書き込みコマンドは、外部端子から供給されたデータを上記書き込み用バッファに書き込み、上記書き込み用バッファに書き込まれたデータは、内部制御回路により自律的にメモリセルに書き込まれるものであることを特徴とするメモリシステム。

【請求項 22】 請求項 10 において、

データは、m本の伝送線のうち、少なくとも n本の伝送線が周期ごとにレベル遷移を行うものであることを特徴とするメモリシステム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

この発明は、半導体集積回路装置、データ処理システム及びメモリシステムに関し、主としてマイクロプロセッサとメモリ装置との間のデータ高速化技術に利用して有効な技術に関するものである。

【0002】

【従来の技術】

メモリコントローラとメモリ間の通信方式は、主として複数本の伝送線路（バス：Bus）を用い、さらに3個以上の装置が物理的に同一の伝送線路で通信を行うことが可能なシェアードバス（Shared Bus）方式が一般的に採用されている。シェアードバスは、1サイクルで複数のデータを通信するため、単位時間当たりの送受信データ量が大きい。さらに装置の数によらずバスは1つでよいので、システムに応じたメモリ量の変更やメモリの追加（増設）を容易にしている。シェアードバスの例として、JEDEC Standard 79, Double Data Rate (DDR) SDRAM Specification（文献1）が挙げられる。

【0003】

【発明が解決しようとする課題】

金属酸化膜半導体トランジスタ（MOS：Metal Oxide Semiconductor）のスケーリングにより、集積回路（IC：Integrated



ted Circuit)、特に中央演算処理装置(CPU:Central Processing Unit)の処理能力は劇的に増加してきた。しかし近年、CPUの処理能力の増加に割にコンピュータシステム全体の処理能力が増加しないという問題点がある。これは、CPUの処理能力に対して、相対的に主記憶(メモリ)の速度が遅くなりつつあることが原因の一つにあげられる。

#### 【0004】

特に、メモリコントローラとメモリ間インタフェースは、上記の通りシェアーバスを採用するため、通信速度の増加に伴い、データ線路間のタイミングのずれ(スキュー)が問題になってくる。また同一線路上に多数の装置が存在することで、装置間のタイミングの違い、装置数による伝送条件の変化、各装置が接続されている地点における信号の反射等の問題が発生する。それにより、特に高速なメモリコントローラメモリ間インタフェースでは、接続されるメモリの数(DIMM:Dual Inline Memory Moduleの枚数)に制限がつく、高価なRegistered DIMMの使用を余儀なくされ、すべてのメモリスロットを使用するとエラーが発生するなど様々な問題点が起こっている。

#### 【0005】

この発明の目的は、高速なデータの伝達を可能とした半導体集積回路装置、データ処理システム及びメモリシステムを提供することにある。この発明の他の目的は、簡単な構成で高速なデータの伝達を可能とした半導体集積回路装置、データ処理システム及びメモリシステムを提供することある。この発明の前記ならびにそのほかの目的と新規な特徴は、本明細書の記述および添付図面から明らかになるであろう。

#### 【0006】

##### 【課題を解決するための手段】

本願において開示される発明のうち代表的なものの概要を簡単に説明すれば、下記の通りである。すなわち、半導体集積回路装置において、命令、情報、情報の所在位置、タイミング信号のいずれかを含む入力信号が供給される入力端子と、上記入力信号に応答し、内部回路で形成された信号又は上記入力端子から供給

された信号を出力させる出力端子とを持つようにする。

#### 【0007】

命令、情報、情報の所在位置、タイミング信号のいずれかを含む入力信号が供給される入力端子と、かかる入力信号に応答して内部回路で形成された信号又は上記入力端子から供給された信号を出力させる出力端子とを備えた半導体集積回路装置の複数個を用い、上記複数の半導体集積回路装置のうち、前段とされる半導体集積回路装置の出力端子と次段とされる半導体集積回路装置の入力端子との対応するもの同士が接続されて縦列形態とし、信号生成回路で形成された上記命令、情報、情報の所在位置、タイミング信号のいずれかを含む入力信号を上記初段の半導体集積回路装置の入力端子に供給し、終段の半導体集積回路装置の出力端子からの信号を上記信号生成回路に供給してデータ処理システムを構成する。

#### 【0008】

コマンド、データ、アドレス、タイミング信号を含む入力信号がそれぞれ供給される入力端子と、上記入力端子から供給された入力信号に対応した信号をそれぞれ出力させる出力端子とを含む半導体記憶装置の複数個を用い、上記複数の半導体記憶装置のうち、前段とされる半導体記憶装置の出力端子と次段とされる半導体記憶装置の入力端子との対応するもの同士が接続されて縦列形態としてメモリシステムを構成する。

#### 【0009】

##### 【発明の実施の形態】

図1には、本発明が適用されたメモリシステムの一実施例のブロック図が示されている。この実施例は、特に制限があるわけではないが、コンピュータシステムにおけるメモリサブシステムに向けられている。図1において、110はメモリコントローラであり、120\_\_0～120\_\_2はメモリモジュール(DIMM)であり、130はメモリチップであり、140はDIMM情報ROMであり、101はクロック伝送線路であり、102はコマンド伝送線路であり、103はデータ伝送線路であり、104はターミネータであり、105\_\_0～105\_\_2はPLL制御信号線であり、106はDIMM情報バスである。

#### 【0010】

図1において、クロック伝送路101、コマンド伝送路102、データ伝送路103の各伝送路は、各メモリチップ130に入力され、その後各メモリチップ130から出力されている。このように接続することで、メモリコントローラ110と直後のメモリチップ130とを接続するクロック伝送線路101とコマンド伝送線路102を除き、対応するもの同士においてすべて1対1の接続となっている。それにより、各伝送線路のタイミング条件が単純化され、さらに素子数の変化による伝送線路の条件変化も起こらず、伝送線路中で信号が反射を起こすことも防ぐことができる。

#### 【0011】

この時、システムクロックをデイジーチェーン接続しないとすれば、クロックーデータ間のタイミングをチップ内で調整するか、もしくは最悪の場合でもデータ転送ができるように規格化しなければならない。本発明では、システムクロックまで含めてデイジーチェーン接続を行っており、すべて2つの素子間のタイミングとして定義される。また信号の伝送方向も一方のみであり、従来のシェアーバスのように双方向通信を行う場合と比較して読み出し／書き込みの切り替えを行う必要のないぶんタイミング条件等が緩和される。

#### 【0012】

ここで、従来のシェアーバス接続と、本発明のデイジーチェーン接続とを比較し、本発明の特徴を明らかにする。本願発明者において、先に検討されたシェアーバス接続におけるメモリコントローラとメモリの接続方法を図27に示し、本願発明に係るデイジーチェーン接続におけるメモリコントローラとメモリとの接続方法を図2に示す。

#### 【0013】

図2と図27において、それぞれ4個のメモリを接続した場合のデータ信号線の接続である。図27と図2において、110\_\_a, 10\_\_bはそれぞれメモリコントローラ、130\_\_a, 130\_\_bはメモリチップ、104はターミネータを示す。さらに、201は伝送線路の寄生容量、202はデータバスの分岐点（スタブ）、203は伝送線路における信号の反射、204, 206は出力ドライバ、205, 207は入力バッファを示す。

## 【0014】

図27のシェードバス接続の問題点として、まず駆動される負荷が大きくなるという点を挙げられる。シェードバス接続では、双方向通信を行うため各メモリコントローラ110\_\_aおよびメモリチップ130\_\_aは、出力ドライバ204と入力バッファ205からなるトライステートバッファを持っている。図27の例において、各出力ドライバ204が駆動しなければならない総容量負荷 $C_a$ は、寄生容量201が $C_p$ 、出力ドライバ204の容量が $C_{da}$ 、入力バッファの容量が $C_{ia}$ とすると、 $C_a = 4 \times C_p + 5 \times C_{da} + 5 \times C_{ia}$ となる。

## 【0015】

この容量 $C_a$ の容量値は非常に大きく、大きな電流駆動力を必要とする。また通信速度が増すと、信号波形の立ち上がり時間／立下り時間を小さくしなければならないため、さらに大きな電流駆動力が必要となる。電流駆動力が大きくなると、出力ドライバ205のサイズが増し、 $C_{da}$ の値がさらに大きくなる傾向にある。そのため、高速な波形の立ち上がり／立下りを得ようとしても、電流駆動力の増加分が出力バッファ205の容量の増加により相殺され、思うように立ち上がり時間／立下り時間が小さくならない。

## 【0016】

次に、信号波形の完全性（シグナルインテグリティ）の問題点を挙げる。シェードバス接続では、1つのバス上に複数のチップが存在するため、バスに必ずスタブ202が存在する。バス配線に直接メモリを実装することで、見た目上スタブ202をなくすことは可能であるが、短いとは言ってもメモリのパッケージ自体がスタブ202となってしまうため、シェードバスにスタブ202は必ず存在するといつてよい。

## 【0017】

このようなスタブ202が存在すると、信号が反射203を起こす。反射203の影響により信号波形が乱れ、信号伝達に支障をきたす。言うまでもないことであるが、通信速度が速くなればそれだけ反射の影響が大きくなる。バスに接続されている機器の個数が一定であるシステムの場合、線路インピーダンスを慎重に設計することで、信号反射を押さえることもできる。しかしメモリサブシステ

ムの場合、ユーザーによるメモリの増設が可能であることという要求が強い。そのため、それぞれ固有の誤差を持つ様々な機器との接続を行う必要があり、信号反射を押さえることは困難である。

#### 【0018】

なお、バスのターミネータ104および、チップ入出力端子においても信号の反射は起こり得る。しかしチップ入出力端子では、スタブ202と比較してインピーダンス整合が取り易く、信号反射の影響は比較的少なくてすむ。これらの理由により、シェアードバス接続を用いて高速通信を行うことは非常に難しくなりつつある。

#### 【0019】

この発明に係るデイジーチェーン接続においては、図2に示した通り、一つの出力ドライバ206が駆動しなければならない総負荷容量 $C_b$ は、寄生容量201が $C_p$ 、出力ドライバ206の容量が $C_{db}$ 、入力バッファの容量が $C_{ib}$ とすると、 $C_b = C_p + C_{db} + C_{ib}$ となる。

#### 【0020】

図2の例において、 $C_b$ の値は $C_a$ と比較して、大体1/4程度になることは明らかである。更に、出力ドライバが駆動しなければならない総負荷容量が減少するため $C_{db} < C_{da}$ となる。それにより、総負荷容量はさらに小さくすることができる。ゆえに、デイジーチェーン接続は、波形の立ち上がり/立下り時間を小さくすることが容易である。

#### 【0021】

さらに、デイジーチェーン接続では信号反射の問題も起きにくい。上でも述べた通り、チップの出力端子ではインピーダンス整合が取り易いため信号反射は起こりにくい。無論、チップの入力端子でもインピーダンス整合を取ることは可能である。さらに、インピーダンス整合を取らなくとも、出力端子側でインピーダンス整合が取れていれば、再び入力端子側に信号が反射してくることはないため、シグナルインテグリティを悪化させる要因とはならない。よって、デイジーチェーン接続ではシグナルインテグリティが良いと言うことができる。

#### 【0022】

このように、シェアードバス接続と比較してデイジーチェーン接続は高速通信を行うのに適した接続法式であると言える。またメモリ増設の容易さにおいても、シェアードバス接続では、バスに接続されたメモリの個数が増えるにしたがって、総負荷容量が大きくなる。またスタブが増えることで信号の反射が大きくなり、シグナルインテグリティが悪化する。一方、デイジーチェーン接続では、メモリの個数が増えても各信号線の条件は変化しないため、メモリ増設も容易である。

#### 【0023】

一般的なシステムでは、特にクロック信号は1つのチップから出力されたものを、他すべてのチップで共有するというアプローチをとることが多い。本発明におけるデイジーチェーンメモリバスシステムではクロック101まで、デイジーチェーン接続を行う。これは、タイミング条件の簡素化が目的である。すなわち、あらゆる信号には必ずタイミングのずれが存在するため、メモリサブシステム全体を規格化するに当たっては、このタイミングのずれも考慮に入れなければならない。

#### 【0024】

ここで、クロック供給と、コマンド、データの供給とを別系統とすると、クロックデータ、コマンド間のタイミングのずれを考慮する必要がある。このタイミングのずれは、ずれの大きさが動作中に大きく変化することも考慮に入れなければならない。このタイミング変化を許容できるようにするためには、コマンド、データの入力から出力までの遅延に、あらかじめマージンをとっておくか、もしくは、ヘッダやフッタにウェイトを挿入したり削除したりしてタイミング調整をできるようにする必要がある。

#### 【0025】

入力から出力への遅延にマージンを持たせておく方法は、有効であるが、マージン分の遅延が直列に接続されるメモリチップ130の個数分積み上がる。これにより、メモリコントローラが読み出し命令を発行した後、データを受け取るまでの遅延時間が大きくなってしまうという問題がある。また、上記ウェイトの挿入／削除はあらかじめ削除可能なウェイトを挿入しておく必要があり、バス効率

が悪化する。上記ウエイトを削除した slot から更に後ろのチップでウエイトを削除することを防ぐ必要があるなど、チップの構成が複雑化する。このようにクロックを別に供給するのは、様々な問題点があり実用的ではない。よって本発明ではクロック 101 も含めて、デイジーチェーン接続する。

#### 【0026】

図3には、この発明に係るメモリチップ130およびその信号線が示されている。同図は、デイジーチェーンメモリバスシステムの信号の内容を説明するものである。各信号線におけるメモリチップ130への入出力を明確にするため、それぞれ、入力クロック101\_\_i, 出力クロック101\_\_o, 入力コマンド102\_\_i, 出力コマンド102\_\_o, 入力データ103\_\_i, 出力データ103\_\_oとする。特に制限するものではないが、この説明では、各クロック101は1bit, コマンド102は8bit, データ103は6bitとする。

#### 【0027】

図3には、この発明に係るメモリチップ130の動作の一例を説明するための波形図が示されている。メモリチップ130のすべての信号はスロット (slot) という単位で区切られる。クロック101は各チップにシステムクロックを供給するという役割のほか、上記のスロットを区切る役割をになう。コマンド102とデータ103は、スロット毎にC0~C7及びD0~D7のような8ワード (word) と、その前後に挿入されたヘッダ (H: Header) とフッタ (F: Footer) とが含まれる。

#### 【0028】

図4に示した通り、メモリチップに入力されたクロック101\_\_i, コマンド102\_\_i, データ103\_\_iはそれぞれ入力タイミングがずれている可能性がある。各チップ内部でそれぞれのタイミングを調整して外部へ出力する。そのことにより、デイジーチェーンメモリバスシステムのタイミング条件は、2チップ間 (出力チップー入力チップ) のみに限定される。

#### 【0029】

図4及び図5には、この発明に係るメモリチップ130間のコマンド/データの伝送方法の概略を説明するための波形図が示されている。言うまでもないこと

であるが、メモリチップ130へは、データのリード/ライト（読み出し/書き込み）を行う必要がある。コマンド伝送路102には、リード命令、ライト命令のほか、行アドレス、列アドレス、バンク指定等の内容が含まれている。これらコマンド情報は、メモリコントローラ110から、メモリチップ130へと一方的に送信される情報であるため、メモリチップ130側は受信するのみであるが、データはメモリチップ130から送信する場合がある。

#### 【0030】

データの送受信は、スロット（slot）単位で行われる。メモリコントローラ110からリード命令が発行されたとき、図5の501に示した通り、メモリチップ130はslotのデータ部をリードデータへ置き換える。すなわち、該slotにおける入力データ103\_\_iはDATA1で示されるデータ集合であるのに対し、該slotにおける出力データ103\_\_oはメモリ103からリードされたデータR0で示されるデータ集合に置き換わる。ライト動作時は、ライト命令によって示されたslotから、データを書き込む（図5の502参照）。これにより、メモリチップ130におけるデータのリード/ライトが可能になる。

#### 【0031】

なお、入力コマンド102\_\_iは、変更を加えず、出力コマンド102\_\_oへ出力される。リードデータへと置き換わらなかった入力データ103\_\_iも同じく出力データ103\_\_oへと出力される。なお、命令体系の詳細な説明については後述する。

#### 【0032】

図6には、この発明に係るメモリチップの一実施例のブロック図が示されている。図6において、601はメモリ内蔵のPLL回路であり、602はコマンドサンプリング回路であり、603はコマンドタイミング検出回路であり、604はコマンドラッチであり、605はデータサンプリング回路であり、606はデータタイミング検出回路であり、607はデータラッチであり、608はデータデコーダであり、609はデータエンコーダであり、610はコマンドデコーダであり、611はマルチプレクサであり、612はクロック生成回路であり、6



13はコマンドパラレルーシリアル変換回路であり、614はデータパラレルーシリアル変換回路であり、615 (615\_\_0~615\_\_7)はバンクFIFOであり、616 (616\_\_0~616\_\_7)はメモリアレイであり、617 (617\_\_0~617\_\_7)はバンクであり、618はモードレジスタであり、620はクロック選択回路である。

#### 【0033】

入力クロック101\_\_iはPLL回路601へ入力され、図7(A)で示されるような30相クロック651 (651[0]~651[29])を生成する。生成された30相クロック651によって、入力コマンド102\_\_iをコマンドサンプリング回路602によりサンプリングする。この時、入力コマンド102\_\_iと各クロックの位相は図7(B)のような関係となる。図7(B)において、0~29の数字はそれぞれクロック651[0]~651[29]の位相に対応している。

#### 【0034】

すなわち、各コマンドは、それぞれ位相がずれた3つのクロックでサンプリングし、合計240個のコマンドサンプル651 (651[29:0][7:0])を得る。ここで[29:0]はサンプリングクロックを、[7:0]は102\_\_iのbit選択を示す。コマンドサンプルを651[3n][7:0], 651[3n+1][7:0], 651[3n+2][7:0]の3つのグループに分ける。

#### 【0035】

すると、それぞれが各コマンドの前半、中央、後半のサンプルとなる。図7(B)の例では、651[3n][7:0]が前半、651[3n+1][7:0]が中央、651[3n+2][7:0]が後半のサンプルである。ただし、前半、後半のサンプルはクロック101やコマンド102のジッタなどの影響により、安定したサンプリング結果が得られないため、中央のサンプルを選択し使用することが望ましい。

#### 【0036】

そこで、コマンドタイミング検出回路603が3つのグループのうち、どのグ

ループが中央のサンプルであるかを判定する。さらに、コマンドラッチ 604 でヘッダとフッタを判定し、ヘッダとフッタを除いた符号化コマンド 653 (653 [7:0] [7:0]) をラッチする。653 [7:0] [7:0] において、前者の [7:0] は、前記図 4 における C0~C7 を示し、後者の [7:0] は入力コマンド 102\_\_i の bit 選択を示す。すなわち C0=653 [0] [7:0] である。

#### 【0037】

データの入力も、ほぼ同じ手順で行う。30 相クロック 651 で入力データ 103\_\_i をデータサンプリング回路 605 でサンプリングし、180 個のデータサンプル 654 (654 [29:0] [5:0]) を得る。データタイミング検出回路 606 で、中央のサンプルを判定する。さらにデータラッチ 607 でヘッダとフッタを判定し、符号化データ 656 (656 [7:0] [5:0]) をラッチする。656 [7:0] [5:0] において、[7:0] は図 4 における D0~D7 を示し、[5:0] は入力データ 103\_\_i の bit 選択を示す。すなわち D0=656 [0] [5:0] である。中央のサンプル及び、ヘッダとフッタの判定法は後述する。

#### 【0038】

特に制限するものではないが、本発明におけるデিজィータリメモリバスシステムでは高速データ通信を行うため、コマンド伝送線路 102, データ伝送線路 103 上のコマンド/データは通信が容易になるように符号化されている。すなわち、これが図 6 の符号化コマンド 653 であり、符号化データ 656 である。よって、入力されたデータをメモリアレイ 616 へ書き込む前に、符号化コマンド及び符号化データを復号しなければならない。

#### 【0039】

符号化コマンド 653 はコマンドデコーダ 610 で復号され、符号化データ 656 はデータデコーダ 608 で復号される。またメモリアレイ 616 から読み出されたデータは、データエンコーダ 609 で符号化される。入力されたコマンドに従って、各バンク 617 へデータを書き込んだりデータを読み出したりする。また、コマンドにはメモリチップ 130 の内部動作等を規定するモードレジスタ

セットがあり、その場合モードレジスタ 618 の内容を書きかえる。

#### 【0040】

また、クロック選択回路 620 は、クロック／コマンド／データ出力用に 30 相クロック 651 から 10 相クロック 662 を選択する。符号化コマンド 653 は、コマンドパラレルーシリアル変換回路 613 で 10 相クロック 662 を用いてタイミングを調整して出力コマンド 102\_\_o として出力される。符号化データ 656 もしくは符号化読み出しデータ 660 は、マルチプレクサ 611 で選択された後、データパラレルーシリアル変換回路 614 で 10 相クロック 662 を用いてタイミングを調整して出力データ 103\_\_o として出力される。

#### 【0041】

マルチプレクサ 611 は、読みだし動作時にメモリチップ 130 からデータを出力する場合、符号化データ 656 の代わりに、符号化読み出しデータ 660 を選択する。出力クロック 101\_\_o は、クロックジェネレータ 612 で 10 相クロック 662 を用いて生成される。この時、クロックジェネレータ 612、コマンドパラレルーシリアル変換回路 613、データパラレルーシリアル変換回路 614 の遅延時間を同じになるように設計すれば、出力クロック 101\_\_o、出力コマンド 102\_\_o、出力データ 103\_\_o の位相が揃う。

#### 【0042】

次に、メモリチップ 130 における実際の読みだし／書き込み動作について説明する。図 8 には、図 6 のバンク 617 内部のブロック図が示されている。バンク 617 は大きく分けて F I F O（ファーストインーファーストアウトメモリ）615 とメモリアレイ 616 とに分かれる。このうち、メモリアレイ 616 は基本的な DRAM と同様な構成をしている。同図において、901 は列アドレス F I F O であり、902 はライトフラグ F I F O であり、903 はライトデータ F I F O であり、904 はリードデータ F I F O であり、905 は行アドレスデコーダであり、906 は列アドレスデコーダであり、907 はメモリセルであり、908 はセンスアンプであり、909 はメインアンプであり、910 はライトバッファである。

#### 【0043】

メモリチップ130における読み出し動作を説明する。図9に読み出し時の各信号の波形図が示されている。ここでは、メモリチップ130のバンク617\_\_4に対するリード命令が行われる。図9のスロット（以下、図面に従ってslotという）0のコマンドCOM0において、バンク617\_\_4の行アドレス（ROW0）指定、バンク617\_\_4の列アドレス（COL0）指定、リード命令（WF0=0）が行われる。次に、図9のslot1のコマンドCOM1において、バンク617\_\_4の列アドレス（COL1）指定、リード命令（WF1=0）が行われる。

#### 【0044】

入力コマンド102\_\_iは、コマンドデコーダ610で復号され、コマンド659として、バンク617\_\_4へ入力される。バンク617\_\_4中で、コマンド659は行アドレス920（ROW0）、列アドレス（COL0）、ライトフラグへ（WF0）と分離される。図9のコマンドCOM0では、行アドレスがバンク617\_\_4を指定しているため、行アドレスデコーダ905が直ちに動作し、選択されたワード線923を立ち上げる。

#### 【0045】

列アドレスもバンク617\_\_4を指定しているため、列アドレスFIFO901へ格納され、ライトフラグWF0はライトフラグFIFO902へと格納される。この段階では列アドレスFIFO901には列アドレスCOL0しか格納されていないため、列アドレスデコーダ906は列アドレスCOL0をデコードする。そして、次の命令COM1で指定された列アドレスCOL1は列アドレスFIFO901へ入力され、ライトフラグWF1はライトフラグFIFO902へ入力される。

#### 【0046】

選択されたワード線923に接続されているメモリセル907はデータ線924とチャージシェアリングを起こし、データ線924に生じた微小なレベル変化は、センスアンプ908で増幅され、メモリセルの内容が読み出される。列アドレスCOL0のデコード結果にしたがって、メモリセル907の内容はメインI/O線925へ出力される。

## 【0047】

ここで、ライトフラグFIFO902から出力されているフラグはリード指定であるため、メインアンプ909はメインI/O線925の内容を増幅し、リードデータFIFO904へ格納する。この段階で、列アドレスFIFO901は列アドレスCOL0の内容を破棄し、次の列アドレスCOL1を列アドレスデコーダ906へ出力する。また、ライトフラグFIFO902も同様にWF0を破棄し、次のWF1を出力する。列アドレスCOL1がデコードされ、メインI/O線にデータが出力されたところで、WF1がまたリード指定であるため、メインアンプ909はメインI/O線925の内容を増幅し、リードデータFIFO904へ格納する。

## 【0048】

図9において、コマンドCOM0が指定されてから、リードデータFIFO904にデータが格納されるまでの時間は、行アドレスが指定されてからtRAC及び列アドレスが指定されてからtCACというスペックで規定される。図9の例では、行アドレスと列アドレスとを同時に指定したため、列アドレスを指定した後、tRAC後にリードデータFIFOにデータが格納される。

## 【0049】

その後、図9のslot2のコマンドCOM2で、バンク617\_\_4に対するFIFO出力命令(OUT)が行われる。すると、メモリチップ130はFIFO出力命令が行われた次のslot3である、図9のslot3で、バンク617\_\_4のReadデータFIFO904の内容を出力する。図9をみてわかる通り、slot3出力時にはR0のデータがリードデータFIFO904の出力に現れているため、図9のslot3における出力データ103\_\_oの内容はR0となる。

## 【0050】

更に、図9のslot3のコマンドCOM3においてもまた、バンク617\_\_4に対するFIFO出力命令(OUT)が行われる。バンク617\_\_4のリードデータFIFO904からは、図9のslot3の出力時にR0が出力されて、図9のslot4の出力時にはR1が出力されている。よって、次のslotで

ある図9の slot 4 の出力データ 103\_\_o の内容は、R1 に置き換わっている。

#### 【0051】

次に、書き込み動作の説明する。図10には、書き込み動作時の各信号の波形図を示す。ここでも書き込み動作と同様にバンク 617\_\_4 に対する書き込み動作が行われるとする。まず、図10の slot 0 におけるコマンド COM0 で、バンク 617\_\_4 に対する行アドレス (ROW0) 指定、バンク 617\_\_4 に対する列アドレス (COL0) 指定、ライト命令 (WF0=1) が行われる。

#### 【0052】

そして、コマンド COM0 のライト命令に対応するデータは、図10の slot 1 のデータ DATA1 として入力される。また、図10の slot 1 のコマンド COM1 では、バンク 617\_\_4 の列アドレス (COL1) 指定、ライト命令 (WF1=1) が行われる。同様に図10のコマンド COM1 のライト命令に対応するデータは、図10の slot 2 のデータ DATA2 として入力される。

#### 【0053】

読み出し動作時と同様、入力コマンド 102\_\_i は、コマンドデコーダ 610 で復号され、コマンド 659 としてバンク 617\_\_4 へ入力される。バンク 617\_\_4 中で、コマンド 659 は行アドレス 920 (ROW0), 列アドレス (COL0)、ライトフラグへ (WF0) と分離される。

#### 【0054】

図10のコマンド COM0 では、行アドレスがバンク 617\_\_4 を指定しているため、行アドレスデコーダ 905 が直ちに動作し、選択されたワード線 923 を立ち上げる。また列アドレスもバンク 617\_\_4 を指定しているため、列アドレス FIFO 901 へ格納され、ライトフラグ WF0 はライトフラグ FIFO 902 へと格納される。

#### 【0055】

この段階では、列アドレス FIFO 901 には列アドレス COL0 しか格納されていないため、列アドレスデコーダ 906 は COL0 をデコードする。そして、次のコマンド COM1 で指定された行アドレス COL1 は、列アドレス FIFO

0901へ入力され、ライトフラグWF1はWriteフラグFIFO902へ入力される。そして遅れて入力された図10のDATA1は、バンク617\_\_4のライトデータFIFO903に格納され、次のDATA2もバンク617\_\_4のライトデータFIFO903に格納される。

#### 【0056】

バンク617\_\_4において、行アドレスROW0及び列アドレスCOL0のデコードが終了する。ここで、ライトフラグFIFO902の出力は、ライト指定のため、ライトバッファ910が動作し、ライトデータFIFO903から出力されているDATA1を、メモリセル907へ書き込む。メモリセル907への書き込みが終了した直後に、ライトデータFIFO903はDATA1を破棄し、ライトフラグFIFO902はWF0を破棄し、列アドレスFIFO901はCOL0を破棄する。

#### 【0057】

次に、列アドレスFIFO901から出力される列アドレスCOL1が、列アドレスデコーダ906でデコードされる。そして、ライトフラグFIFO902から出力されているWF1は、ライト指定(WF1=1)のため、メモリセル907への書き込み準備が出来次第、ライトデータFIFO903からのデータDATA2をメモリセル907へと書き込む。図10では説明を簡略化するために、同じメモリセル907へと書き込むように図示してあるが、もちろん列アドレスCOL0とCOL1が別のアドレスを指示している場合、別のメモリセルへと書き込まれる。

#### 【0058】

メモリチップ130への読み出し／書き込み動作は以上のような手順で行われる。ここでは、行アドレス指定は1回のみ行ったが、別の行アドレスを指定する場合、該バンクにプリチャージ命令を発行した後行アドレスを指定することになる。または、プリチャージ命令を発行せずに行アドレスを指定することも可能だが、その場合はメモリチップ130内部で自動プリチャージがかかる分だけ、リードデータFIFO904へのデータ出力や書き込み終了までの時間がかかる。

#### 【0059】

ここまでは、単一バンクへのアクセス手順を示したが、本実施例におけるメモリチップ130は8バンク構成になっており、各バンクは独立して動作できる。よって、あるバンクがアクティブであっても、別バンクがプリチャージしていれば、行アドレスを指定することができる（バンクインターリーブが可能）。バンクインターリーブを可能にすることで、バスの使用効率を上げることができる。

#### 【0060】

列アドレス指定及びデータ入出力においてFIFOを使用する理由は、DRAMのアクセスの遅さに起因する。一般的に、DRAMへ行アドレスを指定した後、列アドレスを指定し、データが出力されるまでには数十nsの時間がかかる。この値は、半導体プロセスルールの進歩に伴って改善されるが、論理素子の動作速度の改善と比較して、その改善速度は非常にゆっくりしたものであることが知られている。故に、例えばマイクロプロセッサのような論理素子とDRAMとを組み合わせたシステムが、半導体プロセスルールの進歩に伴って進歩した場合、マイクロプロセッサの進歩にDRAMの進歩が追いつかず、システム全体の性能をDRAMが制限してしまうという問題がある。

#### 【0061】

そのため、例えばSDRAM (Synchronous DRAM) では、バンクインターリーブの他に、CASレイテンシ (Latency) CL指定によるバス制御が取り入れられている。すなわち、クロック信号に同期して列アドレスが指定されたのち、CL値で指定されたサイクル後にデータ出力が開始されるという方法である。これにより、行アドレスを変更せず列アドレスのみを変更して行くページモードアクセス時に、前の列アドレスのデータ出力が終了する前に、次の列アドレスを指定することができ、バス使用効率が改善される。

#### 【0062】

しかしこの方式では、CL値がすべてのメモリ素子でそろっていないとバスのコンフリクトが起りやすいため、すべてのメモリ素子でCL値を揃えることが一般的である。これは複数の性能のメモリを混載させた場合、最も性能が劣るメモリ素子に全体の性能が足を引っ張られることを示している。また、一般的なSDRAMで使用されているクロック周波数は66～133MHz程度であるため



、CL値は2～3と比較的小さな値である。しかし今後データレートがあがるに従い、CL値は増えていく傾向にある。CL値が増えていくとメモリコントローラ側の制御が煩雑になってしまう。

#### 【0063】

それに対して、本発明におけるFIFOを用いる方法では、まず列アドレス指定がきわめて自由に行われるということに特徴がある。列アドレスは行アドレスの処理が終わるまで列アドレスFIFO901に格納されるているため、メモリコントローラ110側では、行アドレス処理を考慮する必要が無い。さらに、FIFOであるため、次の列アドレスも自由に投入することができる。

#### 【0064】

さらに、データ出力にもFIFOを用いるためCL制御を行う必要はなく、データがFIFOに格納されるまでの時間のみを考慮に入れば良い。また、CL制御をなくすことで性能が違うメモリを混載しても、高速なメモリは高速に読み出すことができる。データがFIFOに格納されるタイミングが一緒になる可能性はあるが、データ出力命令は1コマンドにつき1つであるため、メモリコントローラ110はコンフリクトの可能性を考慮する必要が無い。

#### 【0065】

さらに、読み出し動作においてはOUT命令の1slot後にデータが出力され、書き込み動作においては、ライト命令の1slot後に、データが入力される。このようにすることにより、メモリコントローラ110は読み出しと書き込みのコンフリクトを容易に回避することができる。つまり、ライト命令とOUT命令を同時に発行しなければ、読み出し／書き込みのデータがコンフリクトを起こすことはない。

#### 【0066】

ただし、前記図1において、データの流れる方向から見て上流のメモリチップ130にライト命令を発行し、下流のメモリチップ130にOUT命令を発行してもデータはコンフリクトを起こさない。逆に、上流のメモリチップ130にOUT命令を発行し、下流のメモリチップ130にライト命令を発行すると、メモリコントローラ110を介さずに、上流のメモリチップ130から出力されたデ

ータが、下流のメモリチップ130へ転送される。これらの特性をうまく使用することで、バス効率の更なる向上をねらうことができる。

#### 【0067】

このように、列アドレス指定およびデータ入出力においてFIFOを使用することで、メモリコントローラ110はデータリードが必要になった時点で自由に列アドレスを投入し、FIFOにデータの準備が出来次第、FIFO出力命令を発行すればよい。つまり、リード命令はメモリセルからFIFOまでのデータ読み出しを指示し、FIFO出力命令がデータ出力命令を指示するという階層的な読み出しとなる。つまり、第1読み出し動作ではメモリセルからFIFOまでの動作が実施され、第2読み出し動作ではFIFOから出力端子に出力されるまでの動作が実施される。また、データライトが必要になった時点で自由に列アドレスを投入し、直後にデータを出力すれば良い。このように本発明におけるデিজィーチェーンメモリバスシステムでは、制御が簡潔でバス効率が高いシステムを構成することができる。

#### 【0068】

次に、コマンドの体系を説明する。特に制限するわけではないが例として、図3を見てわかる通りコマンド伝送路102のバス幅は8bitである。ただし、コマンドには伝送を容易にする符号化が行われているため実際には6bit分の情報を伝えることができる。また図4を見てわかる通り、1つのslot当たりの8ワードの伝送が行われるため、1つのslot当たりのコマンドの情報量は $6 \times 8 = 48$ bitである。特に制限するわけではないが、48bitの内訳は以下の通りである。

#### 【0069】

行アドレスチップセレクト (CSR)	:	3bit
列アドレスチップセレクト (CSC)	:	3bit
出力命令チップセレクト (CSO)	:	3bit
行アドレスバンクセレクト (BSR)	:	3bit
列アドレスバンクセレクト (BSC)	:	3bit
出力命令バンクセレクト (BSO)	:	3bit

命令 (RAS, CAS, OUT, WRI, PRE) :	5 b i t
行アドレス (ROW) :	12 b i t
列アドレス (COL) :	12 b i t
予約 (Reserved) :	1 b i t
合計 :	48 b i t

**【0070】**

行アドレスROW、列アドレスCOL、出力命令OUTは、それぞれまったく独立に命令を発行できるため、チップセレクトとバンクセレクトはすべて専用のb i tが用意されている。命令は行アドレス指定 (RAS)、列アドレス指定 (CAS)、出力命令 (OUT)、ライト指定 (WRI)、プリチャージ (PRE) にそれぞれ1 b i tずつ割り当てる。

**【0071】**

RAS=1の場合、行アドレスチップセレクトCSRで選択されたメモリチップ130における行アドレスバンクセレクトBSRで指定されたバンクの行アドレス (ROW) が指定される。列アドレス (COL) についても同様である。OUT=1の場合、出力命令チップセレクトCSOで選択されたメモリチップ130における出力命令バンクセレクトBSOで指定されたバンクのリードデータFIFO904からデータが出力される。

**【0072】**

ライト指定 (WRI) は列アドレス指定 (CAS) と組み合わせて書き込み動作であることを示す。プリチャージには指定バンクプリチャージと、全バンクプリチャージの2種類の命令が用意される。さらに、RASとBSRの組み合わせで、オートリフレッシュ、セルフリフレッシュ、モードレジスタセットが指定される。

**【0073】**

RAS=1, PRE=1, BSR=任意	:	指定バンクプリチャージ
RAS=0, PRE=1, BSR=3' b000	:	全バンクプリチャージ
RAS=0, PRE=1, BSR=3' b001	:	オートリフレッシュ
RAS=0, PRE=1, BSR=3' b010	:	セルフリフレッシュ

RAS=0, PRE=1, BSR=3' b100: モードレジスタセット

【0074】

上記いずれの場合も、チップセレクトはCSRで指定する。指定バンクプリチャージの場合は、BSRで指定されたバンクをプリチャージする。ReadデータFIFO904の内容は特に操作しない。リード／ライト命令の処理が完了していない場合は、リード／ライト処理を優先する。ゆえに、列アドレスFIFO901、ライトフラグFIFO902、ライトデータFIFO903の内容はプリチャージ時には存在しない。

【0075】

全バンクプリチャージの場合、CSRで指定されたメモリチップ130の全バンクをプリチャージする。この時、リードデータFIFO904の内容も同時にクリアする。リード／ライト命令の処理が完了していない場合は、リード／ライト処理を優先する。ただし全バンクプリチャージが発行された時点で、リードデータFIFO904の内容は破棄されることになるため、リード処理は無視しても問題ない。全バンクプリチャージ命令は、主にスタートアップ時にメモリチップ130の動作を初期化するために使用する。

【0076】

オートリフレッシュは、メモリチップ130内部のリフレッシュカウンタ（図示せず）で自動的に生成される列アドレスのリフレッシュを行う命令である。この場合、全バンク同時にリフレッシュを行う。リード／ライト命令の処理が完了していない場合は、リード／ライト処理を優先する。リードデータFIFO904の内容は保持する。

【0077】

セルフリフレッシュは、PLL回路601、入出力回路、各デコーダ／エンコーダ等を停止し、メモリチップ130内部のリフレッシュタイマ及びリフレッシュカウンタで自動的にメモリ内容を保持するという命令である。デイジーチェーン接続を行っているため、基本的にデイジーチェーン接続の下流側からセルフリフレッシュ命令を与えないとバス動作が不定になる。セルフリフレッシュからの復帰はスタートアップ手順と同様である。

## 【0078】

モードレジスタセットは、主にスタートアップ時にメモリチップ130にチップセレクト番号を割り振ったり、ドライバ回路の電流駆動力を指定したりするためにモードレジスタ618の内容を書きかえる。レジスタの内容は行アドレスROWで指定する。

## 【0079】

特に制限するわけではないが、本実施例におけるデイジーチェーンメモリバスシステムの起動手順（スタートアップ）は以下のようなものが考えられる。スタートアップについて図1をもとに説明する。デイジーチェーンメモリバスシステムは、各メモリがPLL回路601を内蔵しているため、一般的なSDRAMより起動手順が複雑になる。

## 【0080】

1) メモリコントローラ110を起動し、メモリコントローラ110から出力されるクロック信号101が安定する。この時各PLL制御信号105\_\_0～105\_\_2は1を出力している。そして、DIMM情報バス106を通して、各DIMMの情報をDIMM情報ROMから取得する。

## 【0081】

2) PLL制御信号105\_\_0を1→0へ立ち下げる。これによりDIMM120\_\_0のメモリチップ130のモードレジスタ618がリセットされ、PLL回路601がロックを開始する。なお、リセット直後のモードレジスタ618の内容は、チップ番号=3' b111（最下流）、アドレス出力バッファ電流駆動力=最大、データ出力バッファ電流駆動力=最大である。

## 【0082】

3) DIMM120\_\_0のメモリチップ130中のPLL回路601がロックするまで待機した後、PLL制御信号105\_\_0を1に戻す。その後、全バンクリフレッシュ命令を行い（CSR=3' b111）、次にモードレジスタ618設定を行う。モードレジスタ618の内容はチップ番号=3' b000（最上流）、アドレス出力バッファ電流駆動力=適宜、データ出力バッファ電流駆動力=適宜である。出力バッファの電流駆動力は、マザーボードの設計に依存するため適

宜とする。

【0083】

4) PLL制御信号105\_\_1を1→0へ立ち下げる。これによりDIMM120\_\_1のメモリチップ130のモードレジスタ618がリセットされ、PLL回路601がロックを開始する。DIMM120\_\_1のメモリチップ130の動作は上記と同様である。

【0084】

5) DIMM120\_\_0のメモリチップ130中のPLL回路601がロックするまで待機した後、PLL制御信号105\_\_1を1に戻す。その後、全バンクリフレッシュ命令を行い(CSR=3' b111)、次にモードレジスタ設定を行う。モードレジスタ618の内容はチップ番号=3' b001、アドレス出力バッファ電流駆動力=適宜、データ出力バッファ電流駆動力=適宜である。

【0085】

6) PLL制御信号105\_\_2を1→0へ立ち下げる。これによりDIMM120\_\_2のメモリチップ130のモードレジスタ618がリセットされ、PLL回路601がロックを開始する。DIMM120\_\_2のメモリチップ130の動作は上記と同様である。

【0086】

7) DIMM120\_\_2のメモリチップ130中のPLL回路601がロックするまで待機した後、PLL制御信号105\_\_2を1に戻す。その後、全バンクリフレッシュ命令を行い(CSR=3' b111)、次にモードレジスタ設定を行う。モードレジスタの内容はチップ番号=3' b010、アドレス出力バッファ電流駆動力=ゼロ(停止)、データ出力バッファ電流駆動力=適宜である。最下流のアドレス出力は、ターミネータ104で終端されるため出力バッファが動作する必要はない。

【0087】

8) 前記のように、全メモリチップ130が起動したら、リードデータをメモリコントローラ110が正しく受け取れるように、メモリバス全体のレイテンシを計測する。本発明におけるダイジーチェーンメモリバスシステムは、メモリチッ

プ130内部のレイテンシのばらつきを許容できるため、OUT命令が発行されてから、実際にデータをメモリコントローラ110で受け取るまでのレイテンシを測定する必要がある。これは、単純なテストパターンを使用するだけで良い。図1の例ではデータ伝送路103は4系統用意してあるため、4系統を個別にレイテンシ制御を行う。

#### 【0088】

下流のDIMM120上のメモリチップ130の入力クロック101<sub>i</sub>は、上流のDIMM120上のメモリチップ130から出力される出力クロック101<sub>o</sub>であるため、上流のメモリチップ130に搭載されたPLL回路601が安定してから、下流のメモリチップ130に搭載されたPLL回路601を起動する。

#### 【0089】

また、モードレジスタリセット直後は、すべてのメモリチップ130がチップ番号=3' b111に設定されているため、上流のDIMM120上のメモリチップ130がモードレジスタ設定を終えるまで、下流のDIMM120上のメモリチップ130のリセットを解除してはならない。もし解除すれば、上流と下流のメモリが同じチップ番号(=3' b111)となり、チップセレクトのコンフリクトが引き起こされる。

#### 【0090】

前記の例ではチップセレクトを上流から順に設定する方式を示したが、チップセレクトのコンフリクトを回避するために、メモリチップ130内部で下流に出力するチップセレクト信号の内容をデクリメントするという方法も考えられる。すなわち、メモリチップ130自体は各チップセレクト信号=3' b000の場合、自らが選択されたと判断し、モードセレクト信号に頼らない方法である。

#### 【0091】

例えばメモリコントローラ110がCSR=3' b001を出力したとする。最上流のメモリチップ130は、CSR=3' b001を判定し、自分への命令ではないと判断する。そして、下流のメモリチップへ出力する前にCSR=CSR-1という演算を行い、CSR=3' b000を下流のメモリチップに出力す

る。2番目のメモリチップ130はCSR=3' b000を判定し、自分への命令であると判断し処理を行う。その後下流のメモリチップへ出力する前にCSR=CSR-1という演算を行い、CSR=3' b111を下流のメモリチップに出力する。このようにすることで、チップセレクト信号のモードレジスタ設定をしなくても、コンフリクトを起こすことはなくなる。

#### 【0092】

このように、DIMM120を上流から順に、PLLロック及びモードレジスタセットすることで、すべてのメモリチップ130に適切な設定を行うことができる。なお、DIMM120には表面だけにメモリチップ130が存在するシングルサイドDIMMと、両面にメモリチップ130が存在するデュアルサイドDIMMがある。図1の実施例では説明の簡略化のため、シングルサイドDIMMで説明を行ったが、デュアルサイドDIMMの場合も上流側から設定する。また、各チップセレクト信号(CSR, CSC, CSO)及びチップ番号は3bitであるので、シングルサイドDIMMで8枚、デュアルサイドDIMMで4枚まで対応することができる。

#### 【0093】

図1において、DIMM情報ROM140にはDIMMの容量、シングル/デュアルサイドDIMMの区別、推奨電流駆動力設定、tRAC、tCAC等の情報が書き込まれている。

#### 【0094】

次に、この発明に係るデイジーチェーンメモリバスシステムにおけるコマンド/データの符号化について説明する。本発明は、特にこの符号化方式に制限されるわけではないが、例として符号化方式を説明する。デイジーチェーンメモリバスシステムはシェアードバスを使用しないことで、通信の高速化を行っているが、コマンド/データの符号化で更なる高速化を行う。

#### 【0095】

シェアードバスに限らず、伝送線を多ビット用いる、パラレル伝送方式は、伝送線を1本だけ用いるシリアル伝送方式と比較して、信号伝送の周波数を上げにくいといわれている。その原因の最も大きなものは、各伝送線の信号伝送タイミ



ングのずれ（スキュー）である。このスキューが起こる原因は、いくつか考えられるが、大きな原因としては同時スイッチングノイズの影響が挙げられる。

#### 【0096】

一般的にパラレル伝送方式では、多ビットのデータを、ビット数と同数の伝送線にのせ、データ取り込み用のクロック信号（もしくはストローク信号）を同時に伝送し、クロック信号に同期してデータを送受信する。そのため、データの変化が起こらない場合は、クロック信号のみが遷移し、逆にすべてのデータが逆転する場合は、クロック信号を含めすべての信号が遷移する。ゆえに、16 bit パラレル伝送では、クロックのみ1ビットの遷移から、17ビットの同時遷移まですべての可能性が起こり得る。

#### 【0097】

図11に示した通り、ICのパッケージにおいて、各ピンには必ずインダクタンス成分1201が含まれる。信号遷移が起こり出力ドライバ1202に電流が流れると、このインダクタンス成分の影響でIC本体に印可される電源電圧が減少する。電源電圧の減少により、出力ドライバ1202の電流駆動力は減少する。この時、図12に示した通り、1ビット（bit）のみ遷移した場合と、例えば17ビット（bit）遷移した場合とでは、17ビット同時に遷移した方が電流駆動力の減少が激しい。よって、1ビット遷移した場合と、17ビット遷移した場合では17ビット遷移した場合の方が、信号伝送の遅延時間が大きい。このずれがスキューとなって伝わる。

#### 【0098】

本発明のデイジーチェーンメモリバスシステムでは、アドレス伝送線102の8ビット、データ伝送線103の6ビットのうち、毎データ必ずアドレス伝送線102は4ビット、データ伝送線103は3ビット遷移するようにする。そして、伝送線のレベルではなく、直前のレベルと比較して伝送線が遷移したか／遷移しなかったかに情報を符号化する。そのように符号化を行った場合、1回に送ることができるコマンドとデータの場合の数は、 $8C4 = 70$ と  $6C3 = 20$ と表すことができる。

#### 【0099】

6ビットの場合の数は64、4bビット場合の数は16であるため、コマンド6ビット/ワード、アドレス4ビット/ワードを伝送することができる。図13に実データと伝送線103の遷移の対応を示す。Dataで示される実データに対し、Transmission Codeが与えられる。Transmission Codeが1であった場合、伝送線103のレベルは遷移し、Transmission Codeが0であった場合、伝送線103のレベルは直前の値を保持する。アドレス伝送線102の場合も、Dataの部分が4ビットから8ビットへ、Transmission Codeの部分が6ビットから8ビットへと拡張されるだけで基本的には同様である。

#### 【0100】

図13には、No DataにTransmission Codeが割り当てられている。これは例えば、デイジーチェーンメモリバスシステムをグラフィックメモリに応用する場合有効である。グラフィックメモリの場合、ある一点に描画したいという場合が考えられる。しかし高速DRAMでは高速データ転送を実現するため、バースト転送を行うため、描画したい点以外のメモリも書き換える必要がある。

#### 【0101】

従来は、リード・モディファイ・ライト (Read-Modify-Write) で実現するか、データマスク機能で実現していた。ただ、前者は1回の動作で2回のメモリアクセスが必要なためバス効率が悪い。後者はメモリの制御が複雑になるという問題があった。本発明によるデイジーチェーンメモリバスシステムは、図13に示した通り、No Dataが直接転送できるため、容易にデータマスクが実現できる。

#### 【0102】

この符号化方式では、一回符号化/復号に失敗すると、伝送線のレベルの間違いがその後回復せず、復号方式によってはコマンド、データが伝送されなくなってしまう可能性がある。そのため、本発明におけるデイジーチェーンメモリバスシステムでは図4に示した通り、データをslotという単位に切り分け、データ伝送の復帰点としている。

## 【0103】

前記の通り、slotの初めはH (Header) となっている。このHは、伝送線のレベルがすべてLoレベルとなっている。このHはコマンド、データの内容と異なり、伝送線の遷移ではなくレベルで定義される事に注意する。slotの最初であるHは全bitがLoレベルであるため、C0では8bit中4bitがHiレベル、D0は6bit中3bitがHiレベルとなる。この関係を利用することで、容易にHの位置を検出することができる。

## 【0104】

Hの次には8ワードのコマンド、データ本体 (C0~C7, D0~D7) が続く。このコマンド、データに前記の符号化が行われている。ここで、C7, D7における伝送線のレベルは、伝送されるコマンド、データの内容によって決定され、全ビットがハイレベルであることも考えられる。よって、C7, D7から直接H (全ビット=ロウレベル) へと遷移させると、大きな同時スイッチングノイズが発生する可能性がある。これでは、符号化を行った意味が低下する。

## 【0105】

そこで図4に示した通り、C7, D7とHの間にF (Footer) を定義する。このFは、例えばコマンド伝送線102の上位4ビット (102 [7:4]) 及びデータ伝送線103の上位3ビット (103 [5:3]) を無条件でロウレベルにし、コマンド伝送線102の下位4ビット (102 [3:0]) 及びデータ伝送線103の下位3ビット (103 [2:0]) はC7, D7のレベルを保持する。

## 【0106】

すると、C7からFへの遷移は0~4ビットとなり、D7からFへの遷移は0~3ビットとなる。符号化によって、伝送線の遷移数をデータパターンによらず一定にするという目的からは少し外れるが、遷移数が多くなるよりは、少なくなる方が同時スイッチングノイズ耐性が良くなるため、あまり問題にはならない。

## 【0107】

それでもなお問題であるならば、Fを前記のような単純な方法で生成するのではなく、C7, D7のレベルから生成することで対応することができる。Hのレ

ベルがすべてロウレベルであり、その後1ワード毎に、コマンド伝送線では4ビット、データ伝送線では3ビットの伝送線のレベル遷移が起こる。このことから、偶数回のデータ転送が行われた後は、コマンド伝送線、データ伝送線とも偶数ビットの伝送線がハイレベルである。

#### 【0108】

このことから、C7→F→Hの遷移において、必ず4ビットごとの伝送線の遷移が行われるようにFの値を算出することが可能である。同様に、D7→F→Hの遷移において、必ず3ビットごとの伝送線の遷移が行われるように、Fの値を算出することも可能である。このように、C7、D7における伝送線のレベルから、Fの値を適宜算出することでF、Hも含め、1ワード毎の伝送線レベル遷移の数を常に一定に保つことができ、同時スイッチングノイズ耐性を更に改善することができる。

#### 【0109】

図14には、本発明によるデিজィチェーンメモリバスシステムを用いたコンピュータのマザーボード1501のブロック図が示されている。マザーボード1501上には、マイクロプロセッサ（ソケット）1502、メモリコントローラ110、周辺機器コントローラ1503、I/Oコントローラ1504、周辺機器ドータボード（スロット）1505、外部増設ポート1506、メモリサブシステム1507、グラフィックサブシステム（スロット）1508等が配置されている。

#### 【0110】

マザーボード1501は、マイクロプロセッサ1502から、マイクロプロセッサバス1511を通して、メモリコントローラ110へ接続されている。メモリコントローラ110は、メモリサブシステムをコントロールするための機器であり、メモリサブシステム1507の他、周辺機器コントローラ接続ポート1512を通して周辺機器コントローラ1503へ、グラフィックポート1515を通してグラフィックサブシステム1508へと接続される。

#### 【0111】

周辺機器コントローラ1503は、様々な周辺機器を接続するためのバスであ

る、周辺機器バス1513をコントロールする。周辺機器バス1513には、コンピュータの様々なI/O機器をコントロールする、I/Oコントローラ1504をはじめとして、様々な周辺機器ドータボード1505が接続されている。図14では省略されているが、外部機器は外部増設コネクタ1506に接続され、信号は外部機器ポート1514を通して、I/Oコントローラ1504へ接続される。

#### 【0112】

図1のデイジーチェーンメモリバスシステムは、メモリサブシステムに最適されたものである。図14においては、メモリサブシステム1507に應用されている。しかし、図14におけるマイクロプロセッサバス1511、周辺機器コントローラ接続ポート1512、周辺機器バス1513、外部機器ポート1514、グラフィックポート1515等相方向接続を行うバス/ポートに應用することも可能である。また、グラフィックサブシステム1508にもメモリが搭載されることが一般的であり、このグラフィックサブシステム1508のメモリバスにも、デイジーチェーンメモリバスシステムを應用することが可能である。

#### 【0113】

言うまでもない事ではあるが、本発明ではDIMMを用いてメモリを増設しているが、DIMMを用いずに基盤に直接メモリを実装しても良い。また、チップ内部の伝送線に本発明を適用することも可能である。

#### 【0114】

図14にはコンピュータ用のマザーボードとしての応用例を挙げた。メモリサブシステムとして使用する場合、メモリチップ130をメモリサブシステムに接続するために、DIMM及び、DIMMソケットのしくみが必要となる。シェアードバスの場合は、メモリはスタブを通して接続されるため、DIMM及びDIMMソケットの構造は比較的単純であった。

#### 【0115】

しかし、デイジーチェーンメモリバスシステムは、信号がメモリチップ130内部を通る構造となっているため、DIMMソケット→DIMM120→DIMMソケットという信号の流れとなる。ここで、DIMM120が挿入されてい

い状態では、信号が途中で切り離された状態となり、デイジーチェーンメモリバスシステムが機能しなくなる。一般的には、メモリチップ130が搭載されていない、ダミーのDIMMを挿入し、信号の接続を保つ。しかし、この方式はコストがかかる上、ダミーDIMMをユーザーが保存しなければならない等、使い勝手の面で問題が多い。

#### 【0116】

図15にデイジーチェーンメモリバスシステムにおけるDIMM120の概略図を示す。なお、図16は概略図であるため、いろいろな信号線、電源、DIMM情報ROM140、メモリチップ130の数等が省略されている。メモリチップ130はすべて同様の構造である。

#### 【0117】

図15の例では、メモリチップ130の右側に信号線の入力端子101\_\_i, 102\_\_i, 103\_\_iが配置され、左側に信号線の出力端子101\_\_o, 102\_\_o, 103\_\_oが配置されている。また、左右で同じ高さのピンは、同じ信号の入力と出力という対応になっている。このようにメモリチップ130のピンを配置することで、DIMM120上の配線を簡略化することができる。

#### 【0118】

すなわち、表面 (front side) のメモリチップ130の出力端子と、裏面 (back side) のメモリの出力端子が近い位置に存在するため、この2端子を接続するために単純に基盤にスルーホール1602を用いれば最短距離で接続できる。言うまでもないことであるが、この位置関係で重要なのは、入力ピンと出力ピンが、メモリチップ130の反対側の位置にあるということで、この関係において、左右が右左になろうと上下になろうと下上になろうと、ピンの配置が完全に左右対称 (上下対称) でなくとも問題ない。

#### 【0119】

その上で、DIMM120の信号配線パターン1601を表裏で左右対称にすることで、DIMM120の入力端子1603の信号線101\_\_i, 102\_\_i, 103\_\_iと、出力端子1604の信号線101\_\_o, 102\_\_o, 103\_\_oが同じ位置の表裏という関係となる。また言うまでもないことであるが、こ

で重要なのは、D I M Mの端子1603と1604において、同じ信号が表裏の関係になることで、配線パターンの作り方に制限されるものではない。無論裏表の関係になるものは、信号線101, 102, 103であり、その他の信号線及び電源端子は関係ない。

### 【0120】

前記のようなD I M M120の構造を踏まえた上で、D I M Mソケットの構造例を図16に示す。図16はD I M Mソケットの断面図であり、D I M M120が挿入された状態と、挿入されていない状態の2態が図示されている。まず、D I M M120が挿入された状態では、通常のD I M Mと変わらず、ソケットの端子とD I M M120の端子が接続されている。D I M M120が挿入されていない状態では左右の端子が短絡する。

### 【0121】

ここで前記に記したD I M M120の構造によると、D I M M120の端子は表裏が同一の信号の入出力である。よって、短絡することで、D I M M120が挿入されていない状態でも、信号の接続が途絶えず、デイジーチェーンメモリバスシステムを動作させることができる。図16の例では、D I M Mソケットにおける端子のばねの力を利用した単純なものであるが、動作を確実にするため、D I M M120固定用のレバー、D I M M120挿入時の力などを利用して機械的に端子を短絡させることも可能である。これらの構造により、D I M M120のコストや使い勝手も従来のS D R A Mとほぼ変わらないメモリシステムが構成できる。

### 【0122】

D I M M120は基本的に次段への増設が可能であることを前提にしてきた。しかし、D I M M120は必ずしも次段への増設が必要であるとは限らない。例えば、携帯型コンピュータなどの場合、D I M Mスロットが1つしか搭載しておらず、メモリの増量は増設という形ではなくD I M Mの交換という形態を取るものが多い。その場合、図17に示した通り、ターミネータ104をD I M M120上に搭載することで、D I M Mからクロック信号出力101\_\_o及びコマンド信号出力102\_\_oに対応した端子を省略できる。それにより、D I M Mソケッ

トの面積縮小が可能になる。携帯型のコンピュータでは、あらゆる部品の実装面積を縮小することが求められており、増設のできないDIMM120という選択肢もありえる。

#### 【0123】

もちろん、メモリチップ130の構造まで考えに入れば、図18に示したように、クロック信号101<sub>\_\_o</sub>及びコマンド信号102<sub>\_\_o</sub>が出力されないメモリチップ130<sub>\_\_1</sub>を使用して、図17と等価なDIMM120を構成することができる。この場合、メモリチップ130において、出力端子が製造段階で機能しないようにされていてもよいが、モードレジスタ618もしくは外部ピン設定で、オンチップターミネーションを機能させたり出力バッファを停止させたりしても良い。言うまでもないが、図17、図18に示したDIMM120の構造を用いても、増設可能なデイジーチェーンメモリバスシステムでは、最下流のDIMM120として使用することができる。

#### 【0124】

図19には、本発明に係るデイジーチェーンメモリバスシステムの他の一実施例のブロック図が示されている。図19において、基本的な構成は、図1の実施例とあまり変わらない。ただし、メモリチップ150にPLL回路を内蔵せず、外部PLLチップ160が設けられる。これにより、メモリチップ150のチップ面積と消費電力の低減を実現するものである。

#### 【0125】

デイジーチェーンの前段のDIMM120もしくはメモリコントローラ110から出力されたクロック信号101は、一旦PLLチップ160へと入力され、そこから、DIMMクロック101<sub>\_\_1</sub>として、DIMM120上の各メモリチップ150へとクロックが分配さる。また、次段のDIMM120へクロック101が出力される。

#### 【0126】

この方式では、PLLチップ160における2つのクロック出力101<sub>\_\_1</sub>、101の位相差がメモリチップ150における入力クロック101<sub>\_\_1</sub>と出力コマンド102、データ103の位相差と同じであるように遅延量を設計すること



で、クロック伝送線 101 とコマンド伝送線 102, データ伝送線 103 との位相差を小さくすることができる。もちろん、位相差を別の手段で吸収すること、もしくは吸収する必要が無い場合、特に遅延量を一致させなくとも良い。さらにクロック 101 と DIMM クロック 101\_\_1 とを同一の信号線としても問題なくなる。

#### 【0127】

図 20 には、前記図 19 のメモリチップ 150 の一実施例のブロック図が示されている。また、図 21 には、その動作の一例を説明するための波形図が示されている。図 20 において、2101 はコマンド入力遷移検出回路であり、2102 はコマンドラッチであり、2103 はデータ入力遷移検出回路であり、2104 はデータラッチであり、2105 はコマンドパラレルーシリアル変換回路であり、2106 はデータパラレルーシリアル変換回路である。図 21 では、コマンド 102 の入出力について説明するが、データ 103 の入出力についても同様の方法で処理することができる。

#### 【0128】

この実施例において、前記図 1 の実施例と同様にデিজィチェーンメモリバスシステムにおいてコマンド 102、データ 103 は各ワード毎に伝送線のレベル遷移が起こる。よって、信号線の遷移でタイミングを測定しコマンド入力 102\_\_i, データ入力 103\_\_i をラッチすることができる。まず、コマンド入力遷移検出回路 2101 がコマンド入力 102\_\_i のレベル遷移を検出し、10 相コマンドラッチクロック 2151 を生成する。このコマンドラッチクロックにしたがって、コマンドラッチ 2102 がコマンド入力 102\_\_i をラッチする。そして、DIMM クロック 101\_\_1 に同期して、コマンド出力回路 2105 がコマンド出力 102\_\_o を出力する。

#### 【0129】

データ 103 においても、データ入力遷移検出回路 2103 でデータ入力 103\_\_i のレベル遷移を検出し、生成された 10 相データラッチクロック 2152 で、データラッチ 2104 がデータ入力 103\_\_i をラッチする。そして、DIMM クロック 101\_\_1 に同期して、データ出力回路 2106 がデータ出力 10

3\_\_oを出力する。コマンド102、データ103が入力された後のメモリ150の動作は、前記図1の実施例におけるメモリチップ130の動作と同様であるためここでは省略する。

#### 【0130】

前記図20の実施例において、PLL回路を用いずにコマンド102、データ103を取り込む方法を示した。それを応用することで、図22のブロック図に示すような更に他の実施例が考えられる。前記の各実施例と違い、この実施例では、各メモリチップ170へ共通にクロック信号101が入力されているということである。図23に、図22の実施例におけるメモリチップ170のブロック図が示されている。メモリチップ170の動作は、図20の実施例におけるメモリチップ150と同様な動作を行うため詳細な説明は省く。

#### 【0131】

ただし、クロック101とコマンド102、データ103のタイミングが合っていないためメモリ内蔵DLL2401でクロックのタイミングを変更してから、コマンド出力回路2405及びデータ出力回路2406を動作させる。内部クロック2451が10相クロックではないため、コマンド出力回路2405及びデータ出力回路2406は第2実施例と違うものを採用している。

#### 【0132】

クロックタイミングを調整する回路として、前記の例ではDLL2401を採用した。そのため、クロック信号101は、コマンド102、データ103の転送レートと等しい周波数を持つかもしくは半分の周波数を持つことになる。もし、DLLの代わりにPLLを採用すれば、クロック101はコマンド102、データ103の転送レートの $1/X$ という値を採用することができる。

#### 【0133】

これまでの各実施例では、メモリコントローラ110とメモリチップ130、150、170との信号をコマンド102とデータ103とに分けてきた。ここで、ライトデータをコマンド伝送線102上で転送すると図24のブロック図に示した実施例の形になる。このように構成とすることで、メモリコントローラ110のピン数を削減することができる。またリードデータとライトデータを同時

に転送することで、バス効率を向上させることができる。最上流 D I M M 1 2 0 \_\_ 0 上のメモリチップ 1 3 0 の 1 0 3 \_\_ i ピンは使用しないため、マザーボード上で終端処理を行ってあるが、単純に G N D へ接続しても良い。

#### 【0134】

ここまでの各実施例では、メモリを多段に接続することを前程として議論してきたが、メモリの応用には比較的小容量でデータ転送速度が高いことを要求するものがある。例えば高速マイクロプロセッサ用の外部キャッシュメモリや、グラフィックサブシステム用のメモリ等である。これらの応用例では、メモリを多段に接続する必要はない。このような応用では、メモリチップ 1 3 0 のクロック出力 1 0 1 \_\_ o 及びコマンド出力 1 0 2 \_\_ o のピンが不用である。

#### 【0135】

図 2 5 には、本発明のデ이지チェーンメモリバスシステムの更に他の一実施例のブロック図が示されている。図 2 5 の実施例は、これまでの実施例と違い、コマンド伝送線 1 0 2 がメモリコントローラ 1 1 0 に入力されている。これまでの実施例ではリードデータは、データ伝送線 1 0 3 を通してメモリコントローラ 1 1 0 に入力されていたが、図 2 5 の実施例では、データ伝送線 1 0 3 に加えて、不用になったコマンド伝送線 1 0 2 も使用する。

#### 【0136】

上記コマンド伝送線 1 0 2 にコマンドを出力するかデータを出力するかは、モードレジスタを書き換えて対応する。ライトデータは、基本的にデータ伝送線 1 0 3 でメモリチップ 1 9 0 へ入力するが、コマンド伝送線 1 0 2 からデータを送るようにしても良い。このように構成することで、メモリチップ 1 9 0 における読み出し動作時のバス効率が改善される。

#### 【0137】

以上本発明者よりなされた発明を実施例に基づき具体的に説明したが、本願発明は前記実施例に限定されるものではなく、その要旨を逸脱しない範囲で種々変更可能であることはいうまでもない。前記の各実施例では、クロック伝送線 1 0 1 とコマンド伝送線 1 0 2 は、メモリコントローラ 1 1 0 から 1 組出力して、最上流のメモリに分配しており、これらの信号線は 1 対 1 で接続していなかった。

この事で問題があれば、クロック伝送線 101 とコマンド伝送線 102 を必要な分メモリコントローラから出力しても良い。

#### 【0138】

また、図 26 に示した実施例のようにバスバッファ 2701 を利用して分配しても良い。この場合、データ伝送線もバスバッファ 2701 を通せば、クロック 101、コマンド 102、データ 103 のタイミングが一致する。なお言うまでもないことであるが、メモリ 1 個当たりのデータ入出力 103 は 1 組であったがこれが 2 組以上でも問題はない。この発明は、半導体集積回路装置及びデータ処理システム及びメモリシステムに広く利用することができる。

#### 【0139】

本願において開示される発明のうち代表的なものによって得られる効果を簡単に説明すれば、以下の通りである。すなわち、1 つの半導体集積回路装置において、命令、情報、情報の所在位置、タイミング信号のいずれかを含む入力信号が供給される入力端子と、上記入力信号に応答し、内部回路で形成された信号又は上記入力端子から供給された信号を出力させる出力端子とを持つようにすることにより、デイジーチェーン構成で高速なデータの伝達を可能とした半導体集積回路装置を得ることができる。

#### 【0140】

命令、情報、情報の所在位置、タイミング信号のいずれかを含む入力信号が供給される入力端子と、かかる入力信号に応答して内部回路で形成された信号又は上記入力端子から供給された信号を出力させる出力端子とを備えた半導体集積回路装置の複数個を用い、上記複数の半導体集積回路装置のうち前段とされる半導体集積回路装置の出力端子と次段とされる半導体集積回路装置の入力端子との対応するもの同士が接続されて縦列形態とし、信号生成回路で形成された上記命令、情報、情報の所在位置、タイミング信号のいずれかを含む入力信号を上記初段の半導体集積回路装置の入力端子に供給し、終段の半導体集積回路装置の出力端子からの信号を上記信号生成回路に供給することにより、デイジーチェーン構成を利用した高速なデータの伝達を可能としたデータ処理システムを得ることができる。

## 【0141】

コマンド、データ、アドレス、タイミング信号を含む入力信号がそれぞれ供給される入力端子と、上記入力端子から供給された入力信号に対応した信号をそれぞれ出力させる出力端子とを含む半導体記憶装置の複数個を用い、上記複数の半導体記憶装置のうち、前段とされる半導体記憶装置の出力端子と次段とされる半導体記憶装置の入力端子との対応するもの同士が接続されて縦列形態とすることにより、デイジーチェーン構成を利用した高速なデータの伝達を可能としたメモリシステムを得ることができる。

## 【図面の簡単な説明】

## 【図1】

本発明が適用されたメモリシステムの一実施例を示すブロック図である。

## 【図2】

本願発明に係るデイジーチェーン接続におけるメモリコントローラとメモリとの接続方法の説明図である。

## 【図3】

この発明に係るメモリチップおよびその信号線の説明図である。

## 【図4】

この発明に係るメモリチップ間のコマンド／データの伝送方法の一例を説明するための波形図である。

## 【図5】

この発明に係るメモリチップ間のコマンド／データの伝送方法の他の一例を説明するための波形図である。

## 【図6】

この発明に係るメモリチップの一実施例を示すブロック図である。

## 【図7】

図6のメモリチップにおける入力信号の取り込み方法を説明するための波形図である。

## 【図8】

図6のメモリチップのバンク内部の一実施例を示すブロック図である。

**【図 9】**

図 6 のメモリチップにおける読み出し動作を説明するための波形図である。

**【図 10】**

図 6 のメモリチップにおける書き込み動作を説明するための波形図である。

**【図 11】**

この発明が適用される IC のパッケージの等価回路図である。

**【図 12】**

図 11 の IC の動作の一例を説明するための出力波形図である。

**【図 13】**

この発明に係るメモリチップの符号変換の説明図である。

**【図 14】**

本発明によるデイジーチェーンメモリバスシステムを用いたコンピュータのマザーボードの一実施例のブロック図である。

**【図 15】**

この発明に係るデイジーチェーンメモリバスシステムにおける DIMM の一実施例を示す平面図である。

**【図 16】**

この発明に係る DIMM ソケットの断面図である。

**【図 17】**

この発明に係るデイジーチェーンメモリバスシステムにおける DIMM の他の一実施例を示す平面図である。

**【図 18】**

この発明に係るデイジーチェーンメモリバスシステムにおける DIMM の他の一実施例を示す平面図である。

**【図 19】**

本発明に係るデイジーチェーンメモリバスシステムの他の一実施例を示すブロック図である。

**【図 20】**

図 19 のメモリチップの一実施例を示すブロック図である。

**【図 2 1】**

図 1 9 のメモリチップの動作の一例を説明するための波形図である。

**【図 2 2】**

本発明に係るデイジーチェーンメモリバスシステムの他の一実施例を示すブロック図である。

**【図 2 3】**

図 2 2 のメモリチップの一実施例を示すブロック図である。

**【図 2 4】**

本発明に係るデイジーチェーンメモリバスシステムの他の一実施例を示すブロック図である。

**【図 2 5】**

本発明に係るデイジーチェーンメモリバスシステムの更に他の一実施例を示すブロック図である。

**【図 2 6】**

本発明に係るデイジーチェーンメモリバスシステムの更に他の一実施例を示すブロック図である。

**【図 2 7】**

本発明に先立って検討されたシェアードバス接続におけるメモリコントローラとメモリの接続方法の説明図である。

**【符号の説明】**

110…メモリコントローラ、120\_\_0～120\_\_2…メモリモジュール (DIMM)、130、150、170、180…メモリチップ、140…DIMM情報ROM、101…クロック伝送線路、102…コマンド伝送線路、103…データ伝送線路、104…ターミネータ、105\_\_0～105\_\_2…PLL制御信号線、106…DIMM情報バス、

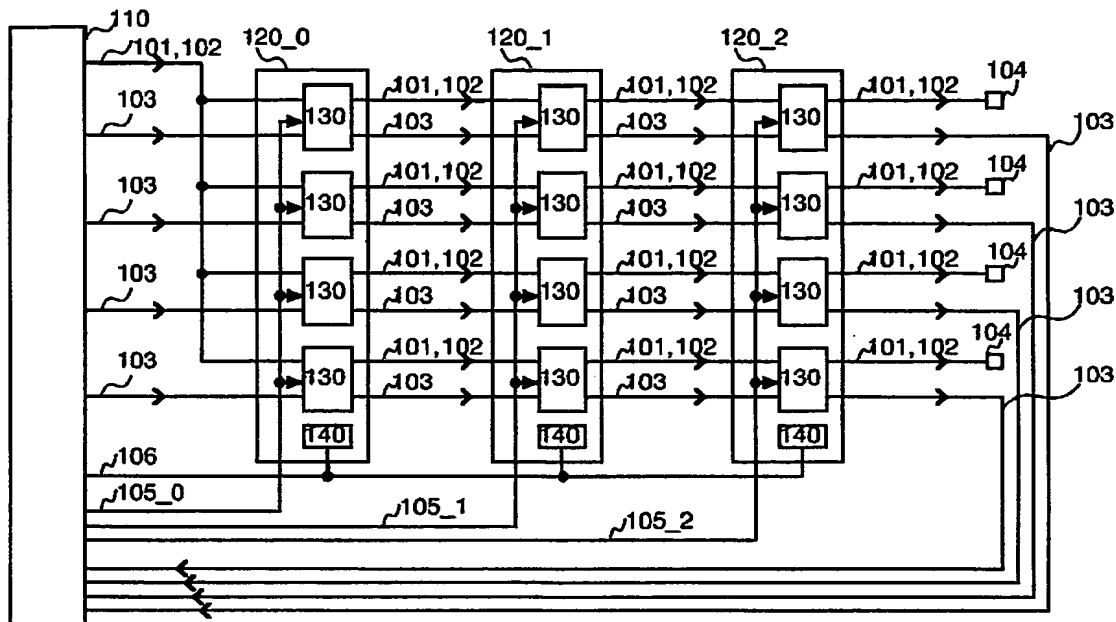
601…メモリ内蔵のPLL回路、602…コマンドサンプリング回路、603…コマンドタイミング検出回路、604…コマンドラッチ、605…データサンプリング回路、606…データタイミング検出回路、607…データラッチ、608…データデコーダ、609…データエンコーダ、610…コマンドデコー

ダ、6 1 1…マルチプレクサ、6 1 2…クロック生成回路、6 1 3…コマンドパラレルーシリアル変換回路、6 1 4…データパラレルーシリアル変換回路、6 1 5 (6 1 5\_\_0～6 1 5\_\_7) …バンク F I F O、6 1 6 (6 1 6\_\_0～6 1 6\_\_7) …メモリアレイ、6 1 7 (6 1 7\_\_0～6 1 7\_\_7) …バンク、6 1 8…モードレジスタ、6 2 0…クロック選択回路。

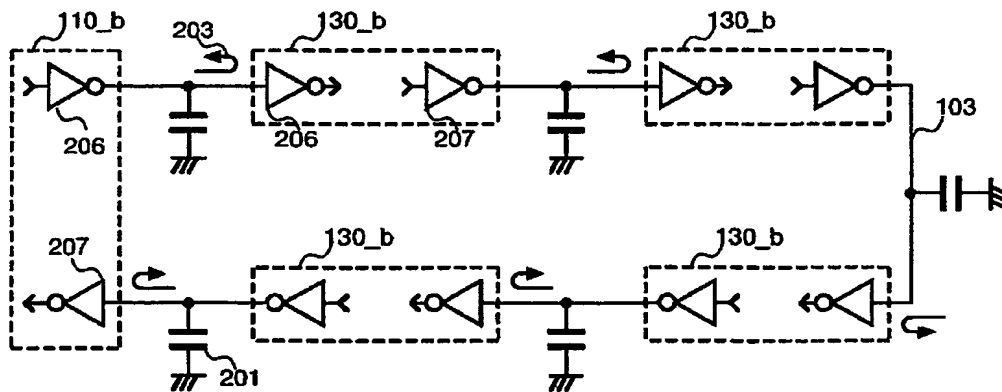


【書類名】 図面

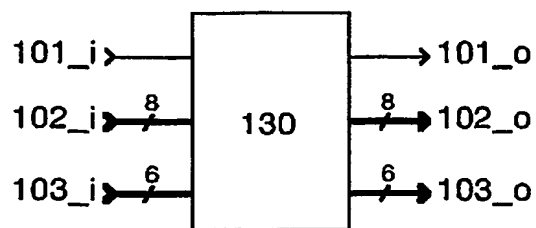
【図 1】



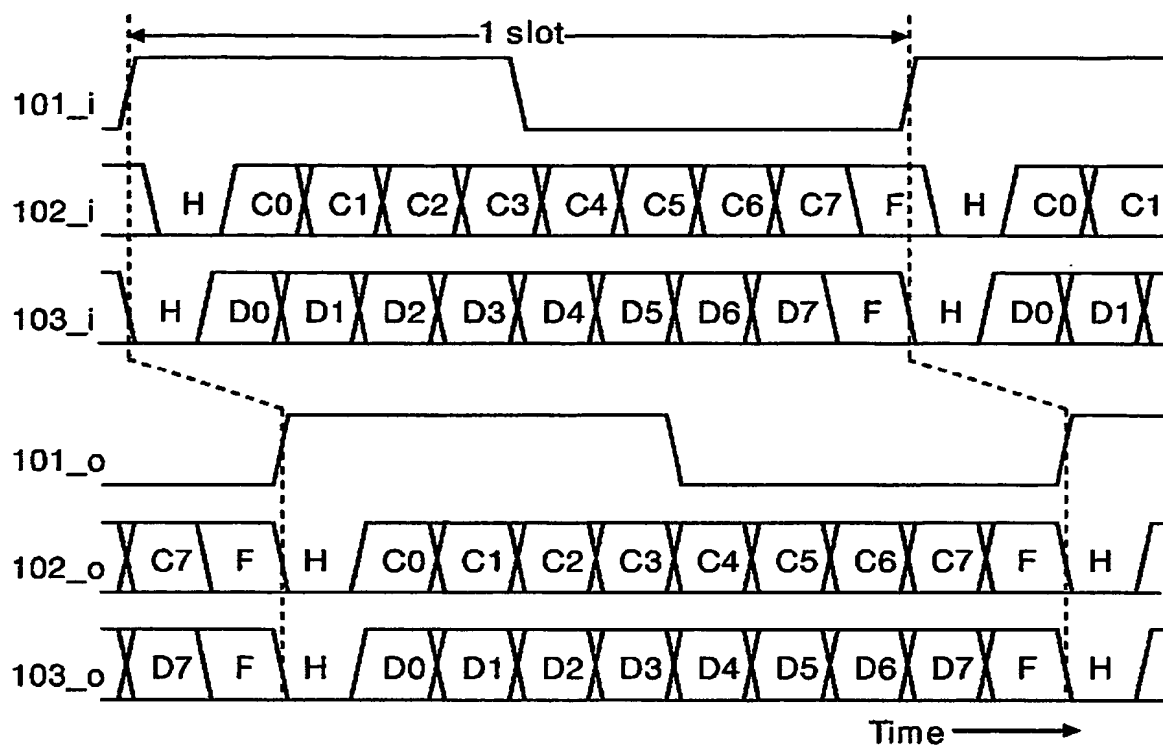
【図 2】



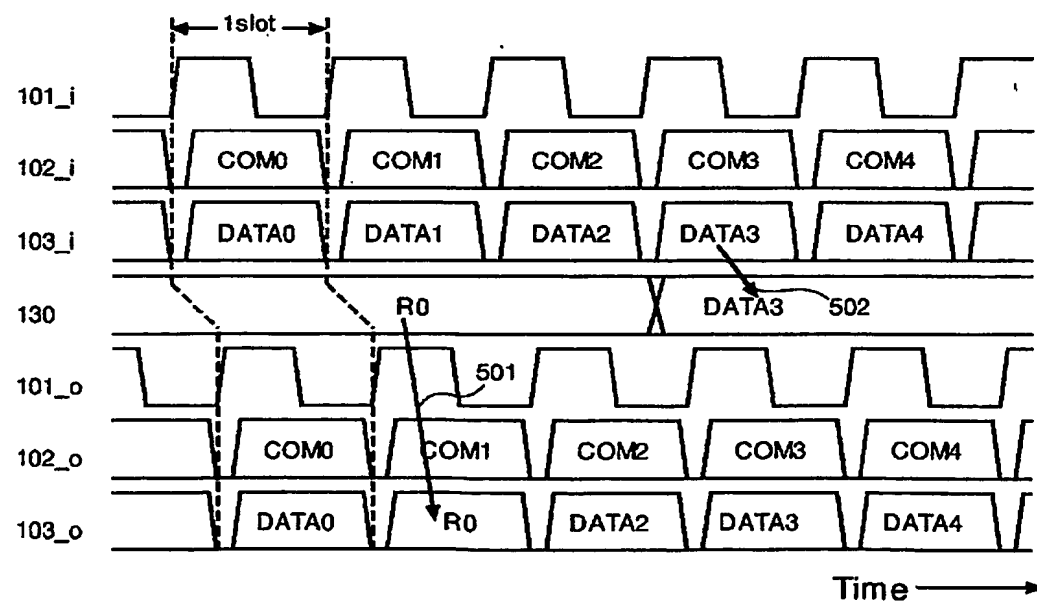
【図 3】



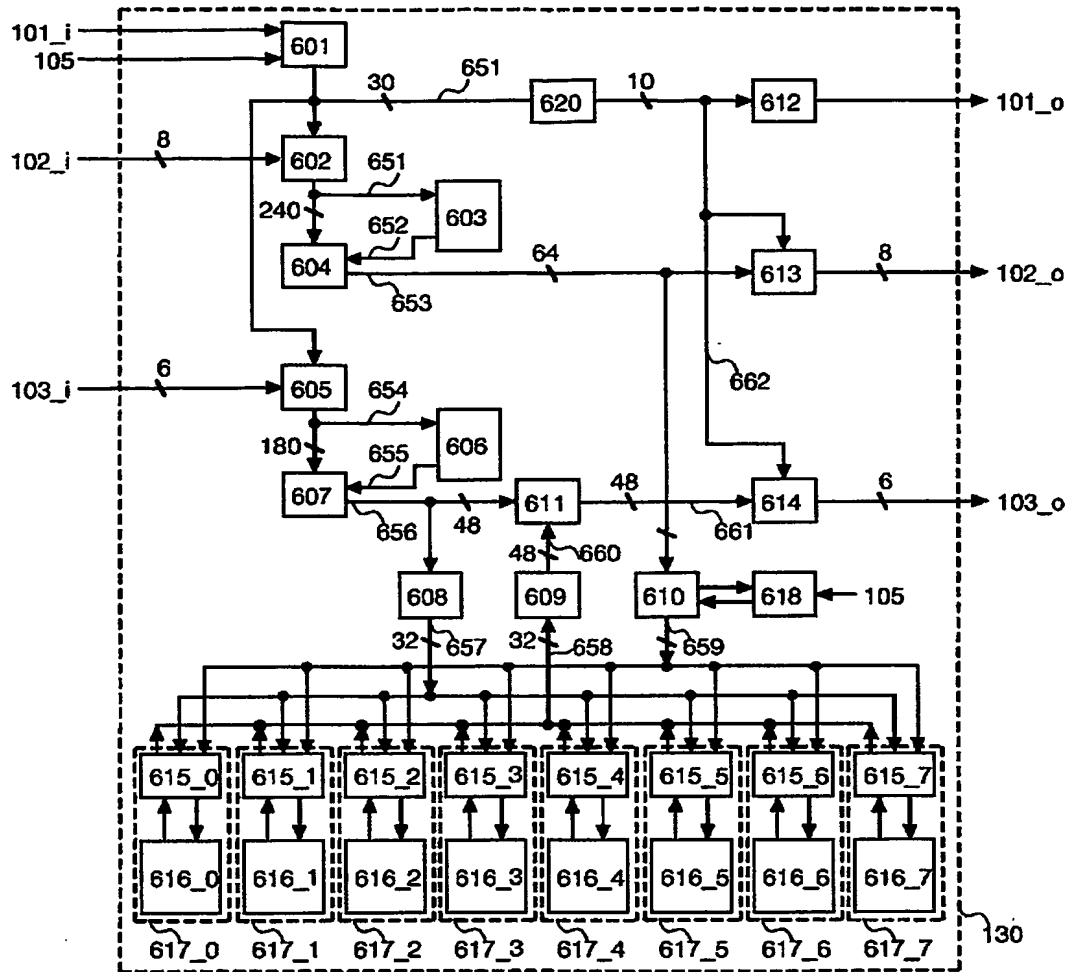
【図 4】



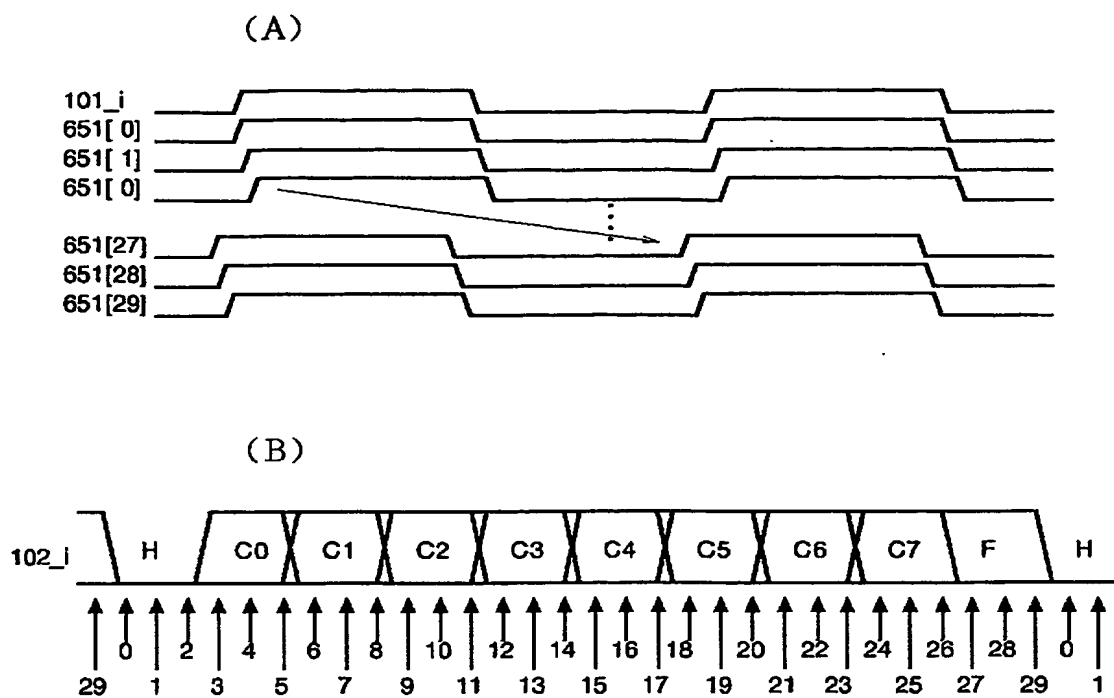
【図 5】



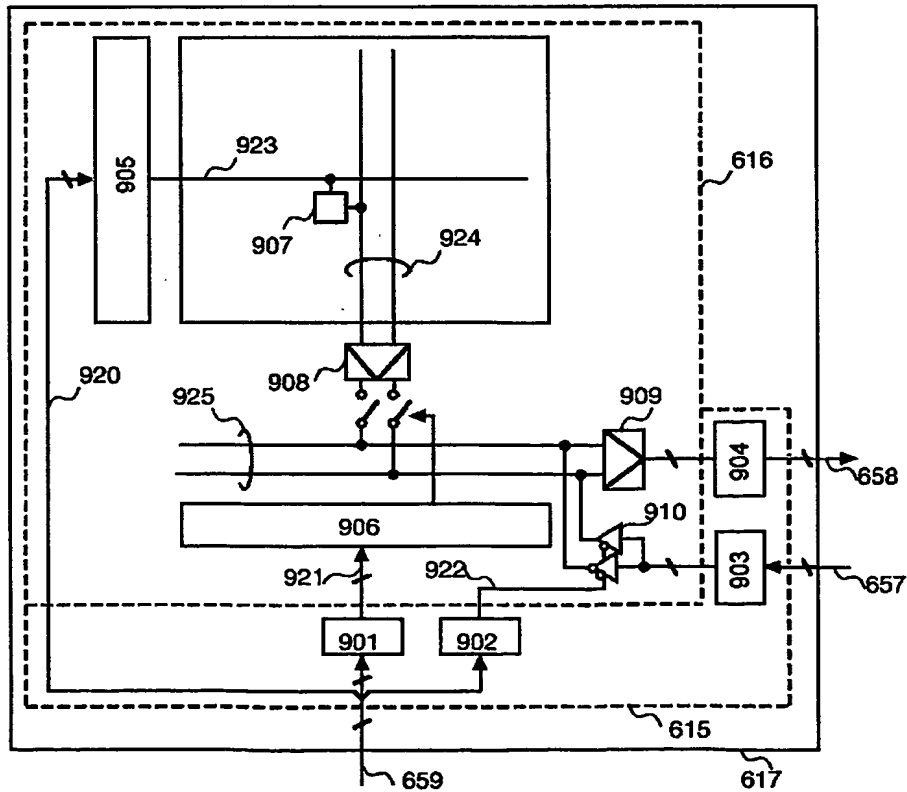
【図 6】



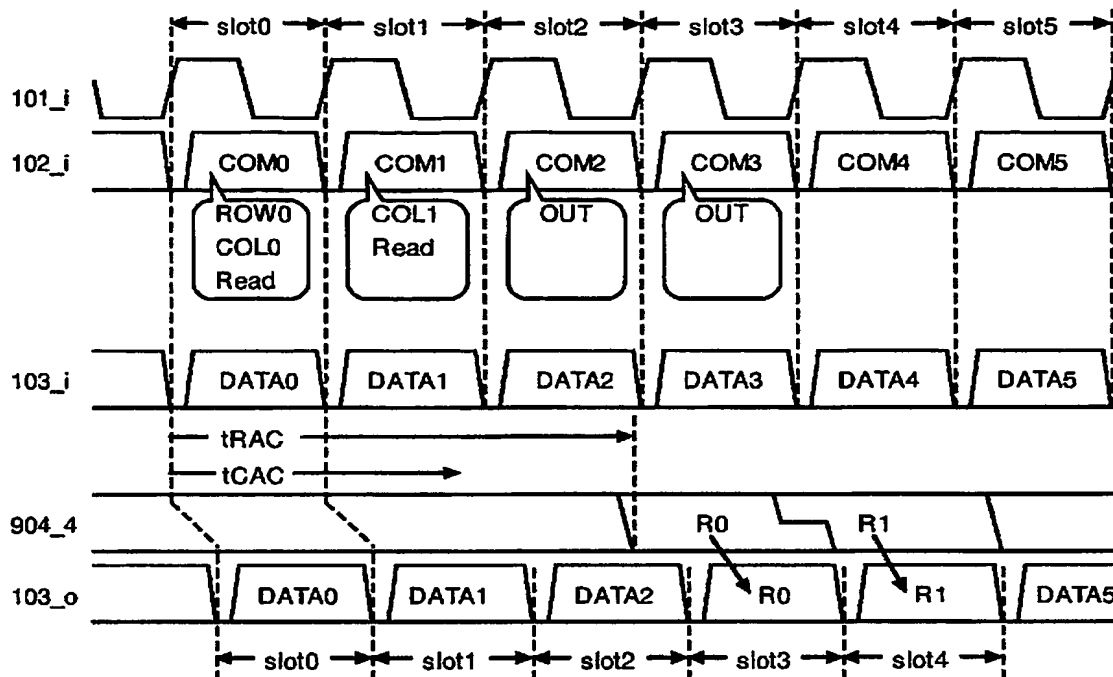
【图 7】



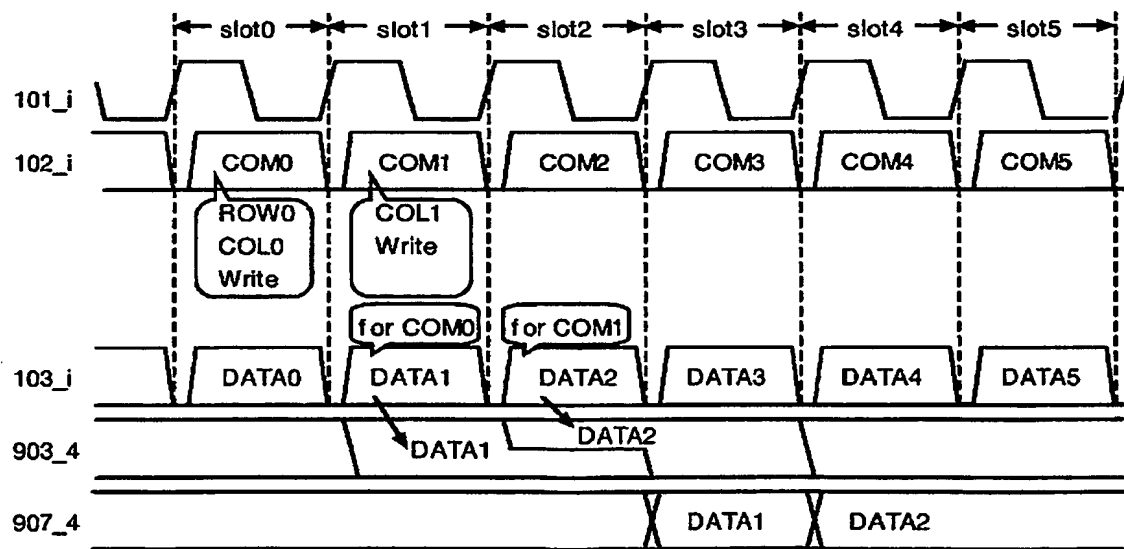
【図 8】



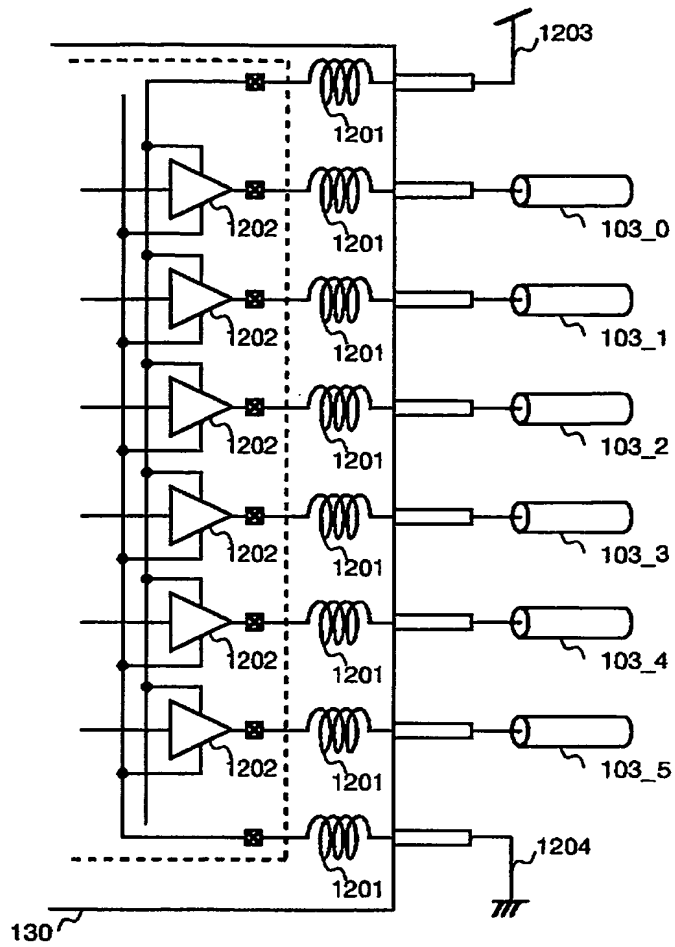
【図 9】



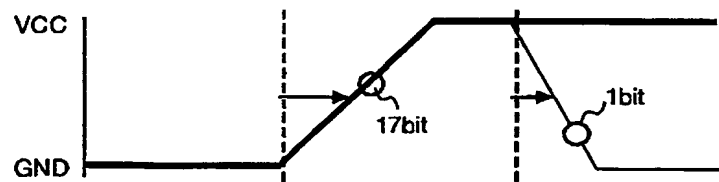
【図 10】



【図 1 1】



【図 1 2】



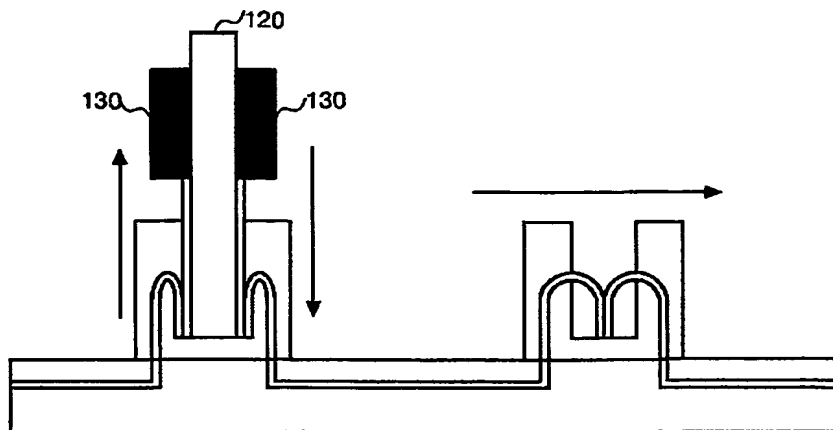
【図 13】

Data				Transmission Code					
0	0	0	0	0	0	1	0	1	1
1	0	0	0	0	1	0	0	1	1
0	1	0	0	1	0	0	0	1	1
1	1	0	0	0	0	1	1	0	1
0	0	1	0	0	1	0	1	0	1
1	0	1	0	1	0	0	1	0	1
0	1	1	0	0	0	1	1	1	0
1	1	1	0	0	1	0	1	1	0
0	0	0	1	1	0	0	1	1	0
1	0	0	1	0	1	1	0	0	1
0	1	0	1	0	1	1	0	1	0
1	1	0	1	0	1	1	1	0	0
0	0	1	1	1	0	1	0	0	1
1	0	1	1	1	0	1	0	1	0
0	1	1	1	1	0	1	1	0	0
1	1	1	1	1	1	0	0	0	1
No Data				1	1	0	0	1	0
				1	1	0	1	0	0
				1	1	1	0	0	0
				0	0	0	1	1	1

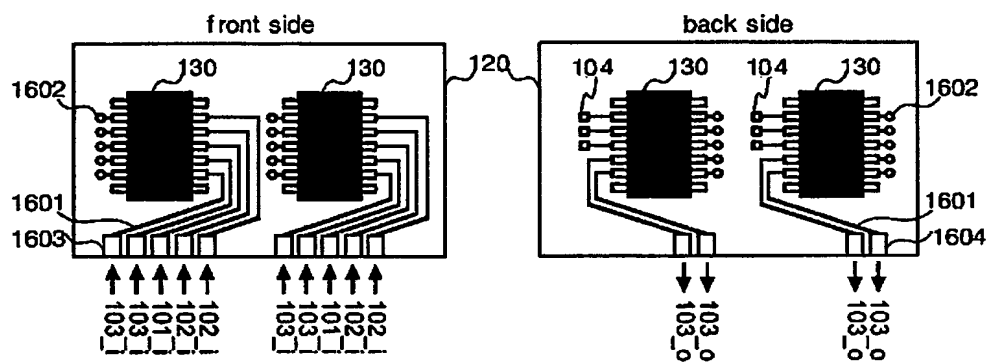




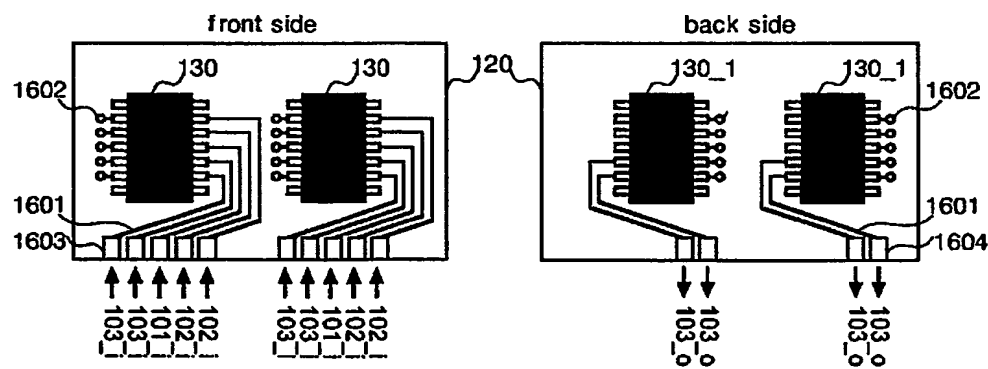
【図 16】



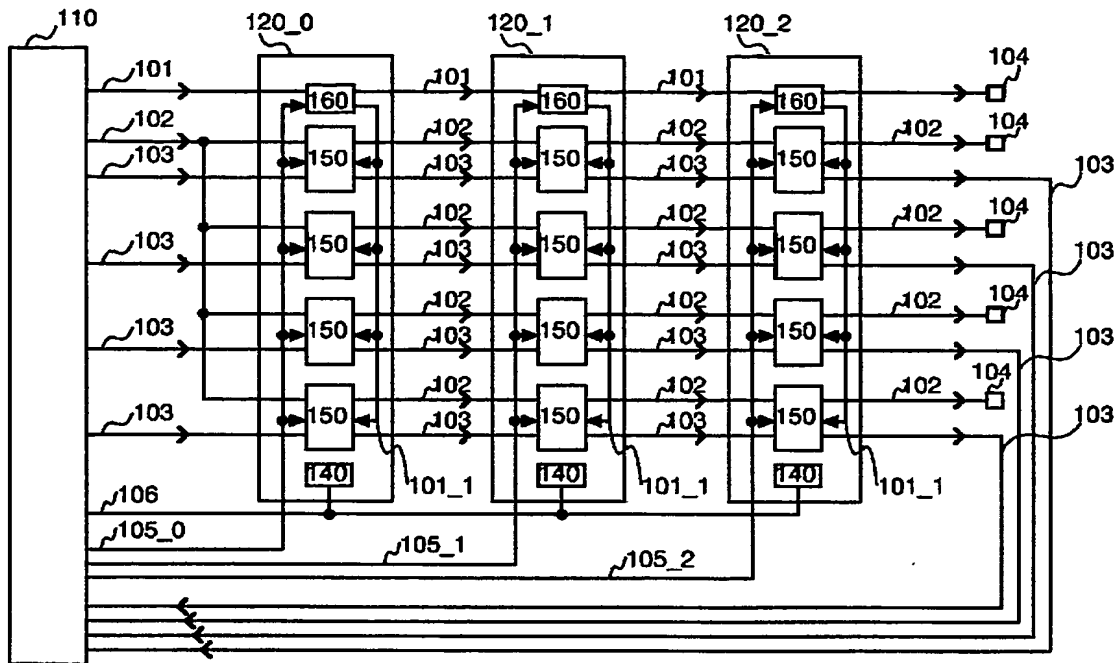
【図 17】



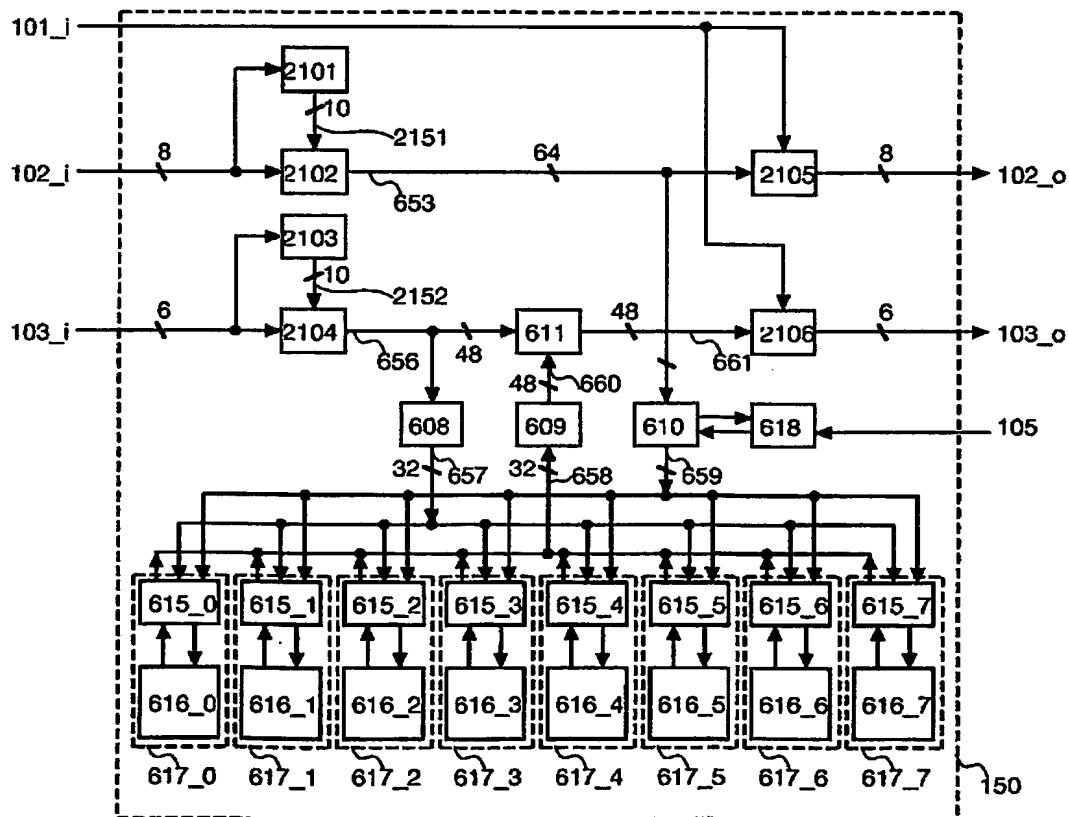
【図 18】



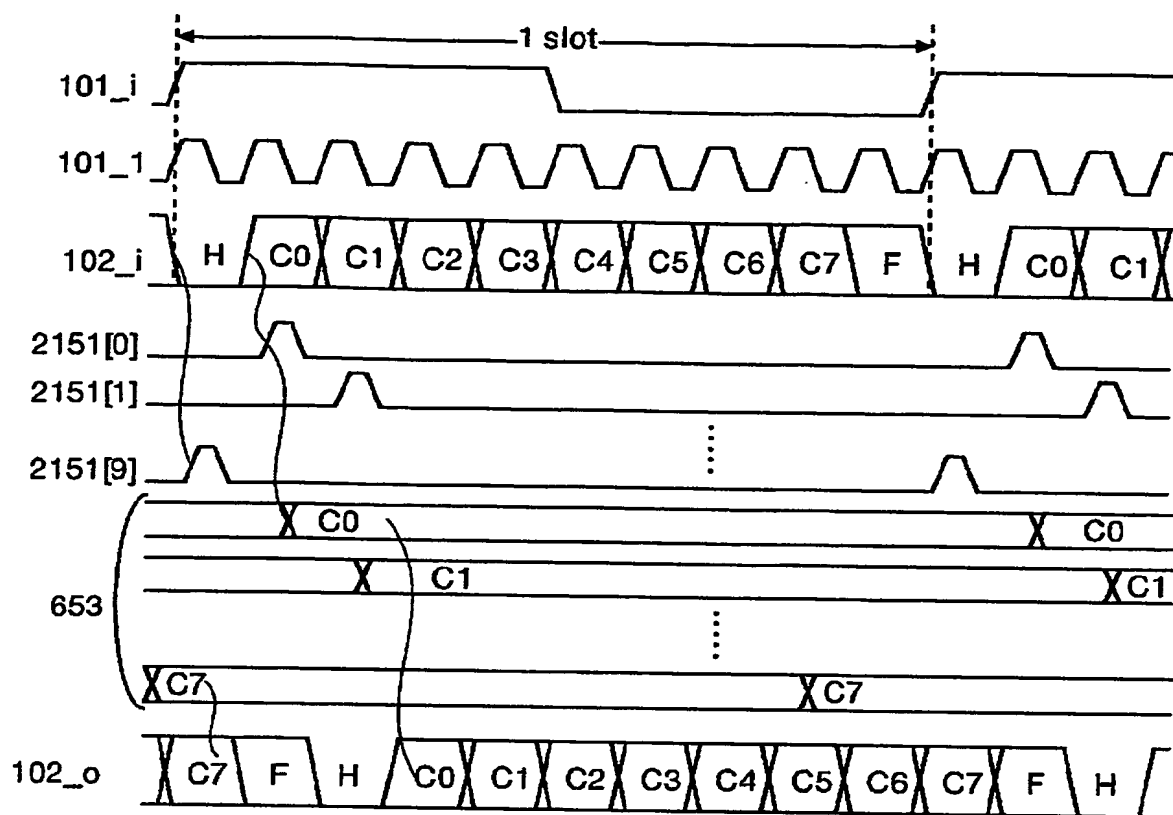
【図 19】



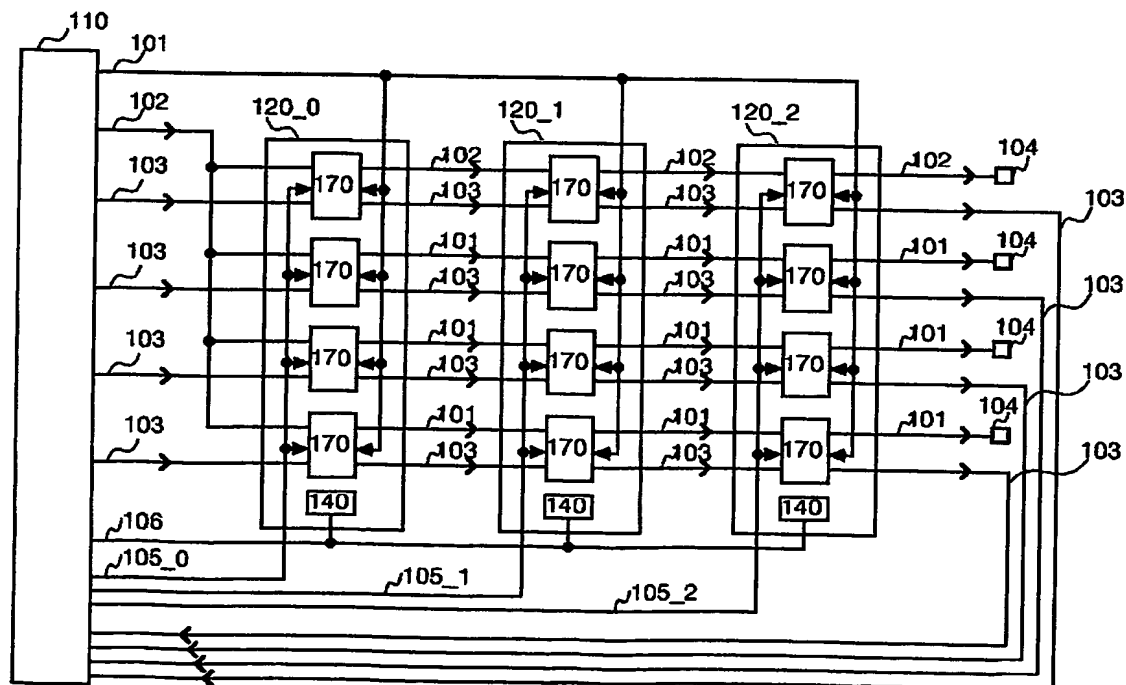
【図 20】



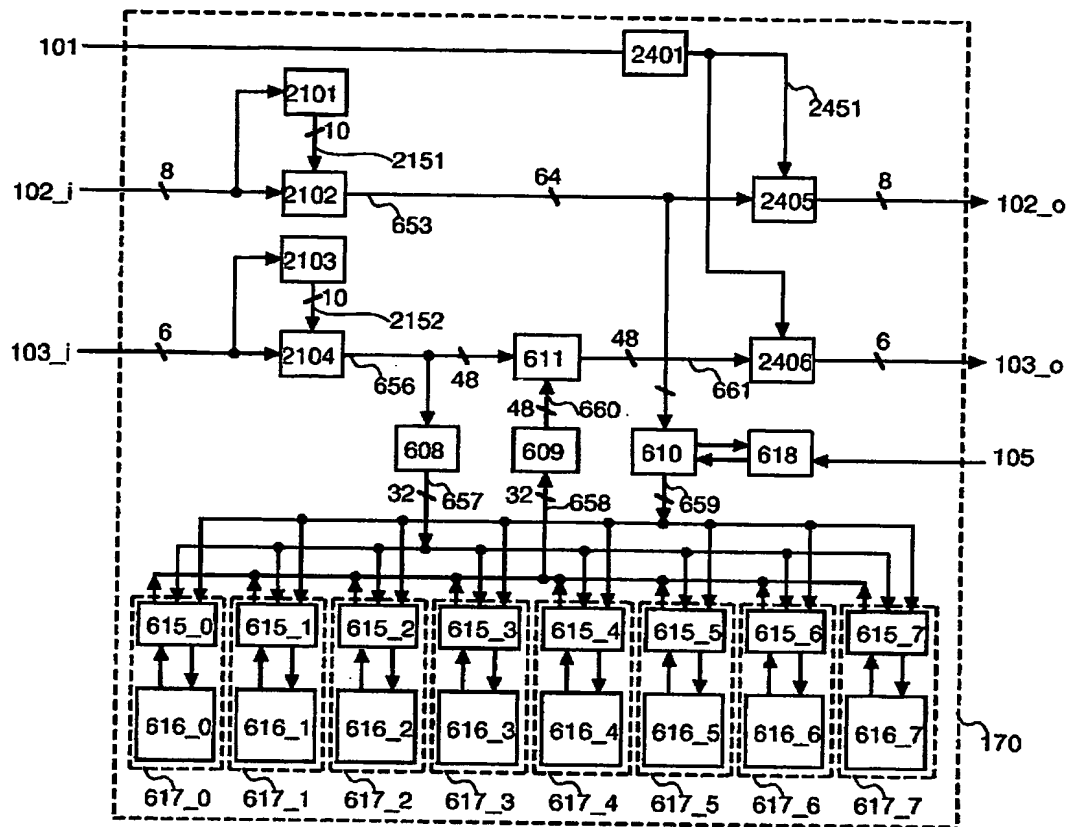
【図 2 1】



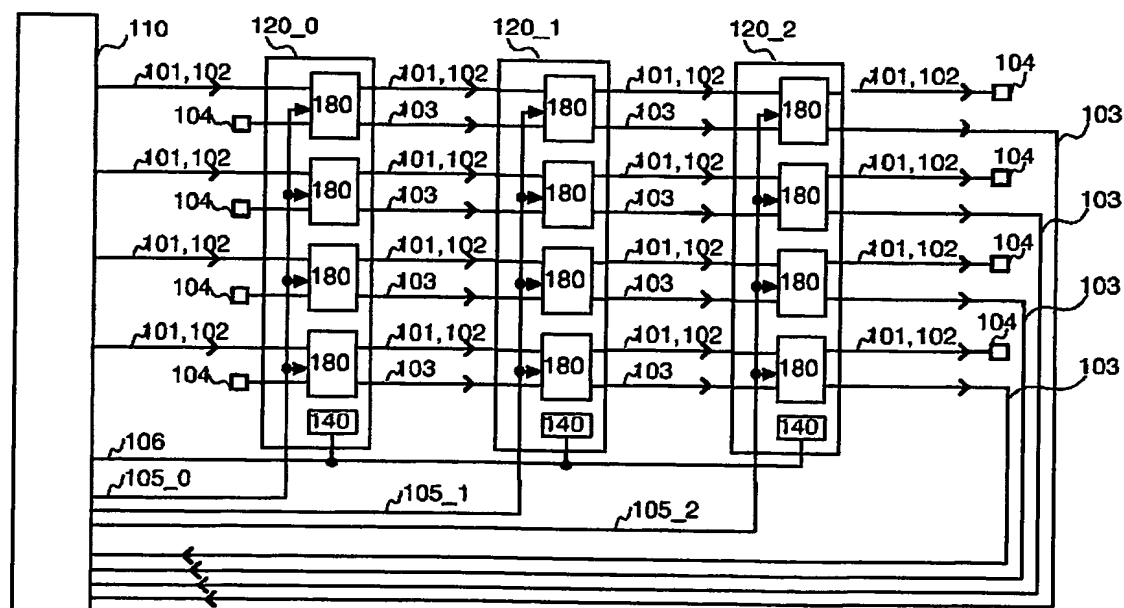
【図 2 2】



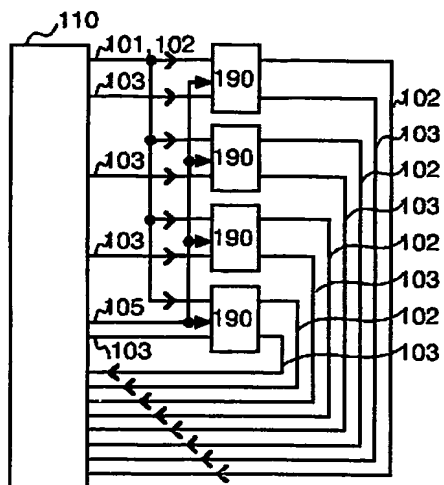
【図 2 3】



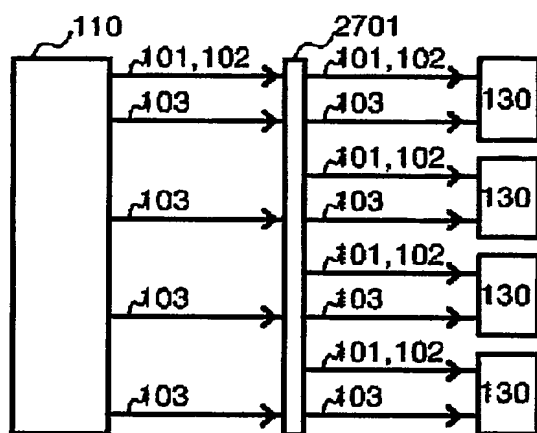
【図 2 4】



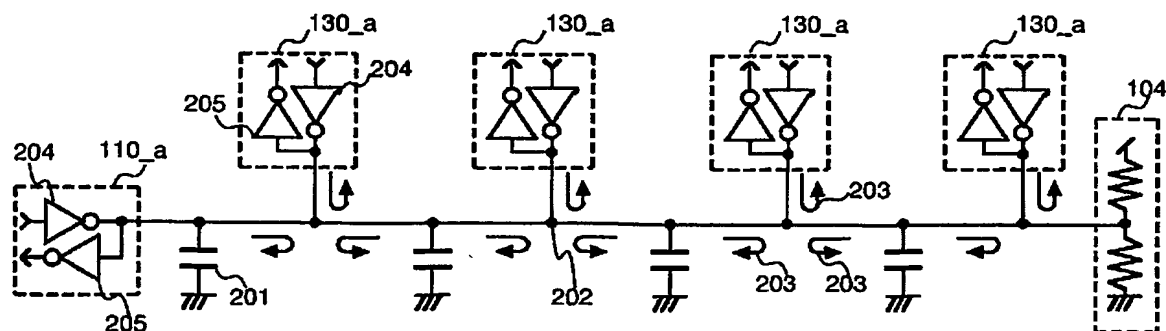
【図 2 5】



【図 2 6】



【図 2 7】



【書類名】 要約書

【要約】

【課題】 高速なデータの伝達を可能とした半導体集積回路装置、データ処理システム及びメモリシステムを提供する。

【解決手段】 命令、情報、情報の所在位置、タイミング信号のいずれかを含む入力信号が供給される入力端子と、上記入力信号に応答し、内部回路で形成された信号又は上記入力端子から供給された信号を出力させる出力端子とを持つ半導体集積回路装置を複数個を用い、上記複数の半導体集積回路装置のうちの前段とされる半導体集積回路装置の出力端子と次段とされる半導体集積回路装置の入力端子との対応するもの同士を接続してデイジーチェーン構成としてデータ処理等のためのデータ転送を行う。

【選択図】 図 1

特願2002-211973

出願人履歴情報

識別番号

[000005108]

1. 変更年月日

1990年 8月31日

[変更理由]

新規登録

住 所

東京都千代田区神田駿河台4丁目6番地

氏 名

株式会社日立製作所

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☒ **BLACK BORDERS**

☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

☐ **FADED TEXT OR DRAWING**

☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

☐ **SKEWED/SLANTED IMAGES**

☒ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

☐ **GRAY SCALE DOCUMENTS**

☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**

☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**